Detecting Novel Discrepancies in Communication Networks

James Abello
DIMACS
Rutgers University
Piscataway, NJ, USA
abello@dimacs.rutgers.edu

Tina Eliassi-Rad
Dept. of Computer Science
Rutgers University
Piscataway, NJ, USA
eliassi@cs.rutgers.edu

Nishchal Devanur
Dept. of Computer Science
Rutgers University
Piscataway, NJ, USA
ndevanur@cs.rutgers.edu

Abstract—**We address the problem of detecting characteristic patterns in communication networks. We introduce a scalable approach based on set-system discrepancy. By implicitly labeling each network edge with the sequence of times in which its two endpoints communicate, we view an entire communication network as a set-system. This view allows us to use combinatorial discrepancy as a mechanism to "observe" system behavior at different time scales. We illustrate our approach, called *Discrepancy-based Novelty Detector* (*DND*), on networks obtained from emails, bluetooth connections, IP traffic, and tweets. DND has almost[1] linear runtime complexity and linear storage complexity in the number of communications. Examples of novel discrepancies that it detects are (i) asynchronous communications and (ii) disagreements in the firing rates of nodes and edges relative to the communication network as a whole.**

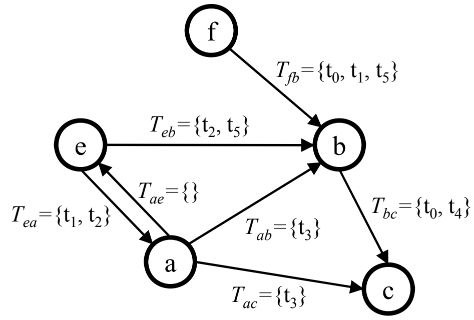


Figure 1. Representation of a communication stream as an edge-labeled network where each edge is labeled by its subset of ground set T of time-stamped communications.

I. Introduction

Given a stream of time-stamped communication pairs $\langle (x, y), t_i \rangle$ (a.k.a. a communication stream), it is useful to find mathematical constructs that "describe" the overall communication behavior of the stream in order to isolate "novel" sub-streams. By labeling each pair (x, y) with its associated time occurrence sequence $\langle t_i \rangle$, we can view a communication stream as a collection of time sequences (one per edge) or as an edge-labeled graph where each edge is labeled by a subset of the ground set $T = \{t_i$ such that t_i is a time-stamp for some pair $(x, y)\}$. Figure 1 depicts this representation for a toy graph. This view of a communication stream (as a collection of subsets of T, one subset per edge) enables us to study time-evolving communication networks as combinatorial set-systems. In particular, we use combinatorial discrepancy as a tool to isolate "novel" communication substreams in a variety of networks including emails, bluetooth connections, IP traffic, and tweets. Our associated algorithm, *Discrepancy-based Novelty Detector* (*DND*), has almost[1] linear runtime complexity in the number of time-stamped edges and linear storage complexity. Across time, our algorithm maintains frequency, firing rate (a.k.a. velocity) and acceleration values for each graph element (i.e., edge, vertex, or a subgraph). At each time t, these values are compared with the corresponding values for the entire network and a tally of the number of ascents/descents in their sequences is used to measure the "discrepancy" of an edge or a vertex with respect to that of the entire network viewed as a set-system.

Our contributions are:

- Introduction of set-system discrepancy as a suitable mathematical notion to study communication networks. To the best of our knowledge this connection has not been explored in the past.
- Introduction of firing rate and acceleration on nodes and edges as interpretable streaming parameters that are useful for communication pattern detection.
- A novelty detection algorithm (DND) based on set-system discrepancy with almost linear runtime and linear storage in the number of time-stamped edges.
- Illustration of the applicability of our approach to a variety of communication networks that in-

[1] The runtime analysis includes an α function, which is a (very slowly increasing) inverse Ackermann's function. For details, see Section III-B.

clude emails, bluetooth connections, IP traffic, and tweets.

The remainder of the paper is as follows. Section II presents related work. Sections III and IV describe our approach and empirical study. Section V concludes the paper.

II. Related Work

The research described here can be broadly placed in the context of anomaly detection [6]. We refrain from using this terminology directly since we use a version of combinatorial discrepancy that is parameterized by particular time-dependent statistics which can be tracked at the edge, vertex, subgraph, or network level. In this setup, what is judged to be "anomalous" according to one particular function may not be "anomalous" with respect to another. We prefer to use the term "novel," which refers to subgraphs whose corresponding communication substream deviate "substantially" from the entire network's communication stream (a.k.a. the system) on a given time interval. This allows us to address two shortcomings of current anomaly detection techniques as pointed in [6]: (1) "When the data has a temporal aspect most of the existing techniques do not handle the sequential aspect explicitly." and (2) "$\cdots$ the nature of anomalies keeps changing over time as intruders adapt their network attacks to evade the existing intrusion detection solutions." Our discrepancy approach implicitly incorporates the potential intruders streams into the overall network statistics and explicitly compares their statistics with the overall time-varying system discrepancy. Substreams that contribute "substantially" to the overall network discrepancy are flagged for further inspection of their communication behavior in relation to the overall system behavior. The same inspection also applies for substreams that do not contribute "substantially" to the overall network discrepancy. Concerted changes of an intruder's communication behavior are expected to be reflected in the overall system behavior.

Approaches to anomaly detection on networked data have explored the use of minimum description length (MDL) principle [16], [5], [18], [9], classification-based methods [15], probabilistic measures [10], spectral methods [13], [12], and neighborhood-based metrics [20], [19], [3]. Aggarwal and Yu [2] investigated outlier detection in high-dimensional data and introduced a solution based on detection by projection. Lee et al. [14] defined outliers as abnormal subtrajectories in motion capture databases. They presented the TRAOD algorithm, whose idea was to partition the trajectories into small segments and then use both distance and density to detect anomalies. [17] proposed a wavelet-based solution to anomaly detection in Border Gateway Protocol (BGP) data. To the best of our knowledge, no approach has been proposed based on combinatorial set-system discrepancy.

Our research can be viewed as part of work on mining frequent subgraphs in dynamic networks [4], [21]. Previous solutions have focused on examining insertions and deletions of edges over time within a fixed population of nodes. Our discrepancy-based approach can handle the insertions and deletions of nodes and edges. Also, previous works in frequent subgraph mining often have parameters such as frequency thresholds and length of patterns that need to be specified by a user. Our method does not require such parameters.

III. Proposed Method

This section presents the overall approach followed by a detailed description of our DND algorithm and its components.

A. Overall Approach

A general approach to the analysis of time evolving communication network is as follows: (a) consider a sequence of graphs $\langle G_t = (V_t, E_t)\rangle$; (b) assign to each edge a time-varying weight W_t; (c) extract at each time t "special subgraphs" consisting of those edges whose weight is smaller than a "suitably chosen" threshold θ. By considering the overall network as a set-system, whose communication pairs have associated accelerations and firing rates,[2] we avoid the need to choose a threshold value and rely instead on a general combinatorial upper-bound on set-system discrepancy. This bound is a function of the number of active communication pairs up to time t (see Section III-B). Those edges or vertices whose acceleration- or firing-rate-sequences deviate substantially from those of the system are flagged as special and are maintained in a union-find data structure (see Section III-B). Since edges come and go depending on the system's activity, we take snapshots of the system at those times for which the overall system acceleration and firing rate are at local maxima. For particular time intervals, firing rates and accelerations are statistics intended to capture bursty communication behavior at an edge, at a vertex, and at the overall system level. Capturing those time intervals on which certain subgraphs "drive" the overall system communication pattern is in our view one of the central questions in understanding time-evolving networks. Set-system discrepancy is a conceptual construct that enables this undertaking. Next section describes our algorithm, called *Discrepancy-based Novelty Detector* (*DND*).

[2] Firing rate can be thought of as velocity.

B. Algorithm Description

Our Discrepancy-based Novelty Detector (DND) represents a time-varying graph[3] as a set-system. Namely, a time-varying graph, on a set of vertices V, is a collection of time-stamped communication pairs $\langle (x,y), t_i\rangle$ where x and y are elements of V and t_i indicates the time-stamp when the edge (x,y) was active. For each edge $e=(x,y)$, we let $T_{e,t}$ denote the set of time-stamps ($t_i \leq t$) in which e is active. $|T_{e,t}|$ is the frequency of communications on edge $e=(x,y)$. We denote the collection of active node-pairs up to time t by $E_t = \{e=(x,y) : T_{e,t} \text{ is non-empty}\}$.

Each edge in E_t has a firing rate: $fr(e,t) = \frac{|T_{e,t}|}{t}$ and its corresponding firing sequence is $fr(e) = \langle fr(e,t)\rangle$. The firing rate of any subset E' of E_t is just the sum of the firing rates of the edges in E' up to time t; and its corresponding firing sequence will be denoted by $fr(E') = \langle fr(E',t)\rangle$. The firing rate of a vertex x (up to a particular time t) is the sum of the firing rates of its incident edges up to that time t. With these conventions, a time-varying graph is a graph sequence $\{G_t = (V, E_t)\}$ with a corresponding firing sequence $\langle fr(E_t)\rangle$. We will refer to $fr(E_t)$ as the firing rate of G_t.

The aforementioned mathematical framework allows us to formulate questions related to the "behavior" of either vertices or edges in a graph sequence $\{G_t\}$ in terms of their associated firing rates. The overall approach consists of comparing the firing sequence of an edge or a vertex with the firing sequence of the graph in which they reside. With this in mind, each G_t can be seen as the set-system: $S_t = \{T_{e,t} : T_{e,t}$ is a subset of a fixed ground set of time-stamps $T\}$. Therefore, a time-varying graph becomes a special set-system; and we adapt tools from set-systems' discrepancy theory [8] to study aspects of its behavior. Next, we introduce the definition of combinatorial set-system discrepancy.

1) Combinatorial Set-System Discrepancy: For the set-system $S_t = \{T_{e,t} : T_{e,t}$ is a subset of $\{t_0, \cdots, t\}\}$ associated with a graph G_t in the time-graph sequence $\{G_t\}$, and any two-coloring function $\chi : T \longrightarrow \{-1, 1\}$, let $\chi(T_{e,t}) = \sum\{\chi(t_i)$ for $t_i \in T_{e,t}\}$. This is called the *discrepancy* of the edge e at time t, with respect to the coloring χ, and abusing notation we denote it by $\chi(e,t)$. The $\chi_{discrepancy}$ of S_t is $\max\{|\chi(T_{e,t})|$ for $T_{e,t} \in S_t\}$. The discrepancy of the set-system S_t is the minimum over all χ of $\chi_{discrepancy}(S_t)$. It follows from a fundamental result in combinatorial discrepancy [8] that the maximum discrepancy of any of our set-systems is less than or equal to $\sqrt{2t' \ln(2m_{t'})}$, where t' is the maximum time when any edge is active and $m_{t'}$ is the overall number of active edges up to time t'. Moreover, a random, uniform and independent coloring of $\{t_0, \cdots, t\}$ achieves this maximum [8]. This provides us a mechanism to associate to each edge e at time t, a χ-weight in the following manner: $\chi\_wgt(e,t) = |(|\chi(e,t)| - \sqrt{2 * t * \ln(2m_t)})|$. The χ-weight of an edge up to time t measures how far is its χ value, $\chi(e,t)$, from the corresponding theoretical upper-bound on discrepancy $\sqrt{2 * t * \ln(2m_t)}$, which we refer from now on as the $sbp(t)$.

2) Novel Edges: An edge e will be called *i-novel* if its $\chi\_wgt(e,t)$ is i standard-deviations away from the mean of the weight distribution $\chi\_wgt(E_t)$. Notice that edges with quite different overall "activity" may have close χ-weights. When this is the case, it can be interpreted as a strong indication that their "activity patterns" are "similar" with respect to χ even though one edge may be vastly more active than the other. When the discrepancy $\chi(e,t)$ is close to 0, it can be interpreted as an indication that e's "activity pattern" may be difficult to detect. On the other hand, edges with high absolute discrepancy (i.e. with low χ_wgt) exhibit an activity pattern different from the activity pattern of the entire system (again with respect to χ).

3) Choosing the Coloring Function χ: Even though there are locally optimal derandomization methods that produce a random coloring χ [8], we opt here for an *Ascents Coloring* ($A\chi$) of T that keeps track of the "recent ascents" of the firing sequence $\langle fr(E_t)\rangle$. Namely, if $pred(t)$ denotes the largest time t_i smaller than t for which there exists an edge e active at time t_i, and if $fr(E_t) \geq fr(E_{pred}(t))$, then we let $A\chi(t) = 1$; otherwise, $A\chi(t) = -1$. With this coloring $A\chi$, the discrepancy $A\chi(e)$ of an edge e at time t measures the amount of agreement or disagreement that the firing sequence of the edge e, $\langle fr(e,t)\rangle$, has with respect to the "ascent" firing sub-sequence of $\langle fr(E_t)\rangle$. Those edges whose discrepancy (with respect to this "ascent" coloring $A\chi$) differs "substantially" from the discrepancy of the sequence $\{G_t\}$ will be called "novel." In some contexts this may be called "anomalous," but we refrain from using this term because what is "judged anomalous" by one coloring χ may not be so under a different coloring. In summary, our task is to find "novel" edges under the coloring $A\chi$. Next we detail our scalable procedure to compute such $A\chi$ novel edges.

4) The DND Algorithm: Given the definitions of combinatorial set-system discrepancy, novel edges, and the choice of coloring function χ, we define our DND algorithm.

- **Input:**
  A streaming graph sequence $\{G_t\}$ consisting of time-stamped communication pairs $\langle (x,y), t_i\rangle$,

[3] We use the terms graph and network interchangeably.

where x and y are elements of V. We allow for t_i to be given explicitly or it can be recorded when the pair (x, y) arrives into a system monitoring communication activity between the elements of V.

- **Output:**
  At each time t, a collection of novel edges Sp_t is produced which are marked as i-novel, where i is the number of standard deviations the current $\chi\_wgt(e, t)$ deviates from the mean. Each Sp_t is accompanied by a list of metrics (see Table II).
- **Algorithm:**
  1) ⟨*Initialization*⟩
     a) Initialize a Union-Find Data Structure [11] to maintain a Minimum Spanning Forest of the weighted graph $\langle V, E_t, FR_t \rangle$ where FR_t assigns to each existing edge e its firing rate $fr(e, t)$ and $acc(e, t)$, where $acc(e, t) = \frac{1}{\Delta(t)} \times (fr(e, t) - fr(e, t - \Delta(t)))$. $\Delta(t)$ is the time-difference between the current and last occurrence of e. We assume the system has enough memory of size $k \times |V|$, where k is a constant equal to the number of bytes required to store a vertex id, the firing rate of an edge, and the last time the edge was active.
     b) Set-up k fixed size *buffers*[i] for output.
  2) ⟨*Updating the Minimum Spanning Forest*⟩: Read into an input buffer B a sequence of arriving edges,
     a) Update the firing rate of G_{t+buff}, where $t + buff$ depends on the last active time of any edge in the buffer B. Similarly for acceleration.
     b) Compute $\chi\_wgt(e, t)$ and place into output *buffer*[i] those edges whose weights have a value i standard deviations from the mean of the current edge weight distribution. Determine which edges in the buffer are replacement edges (i.e., if there are edges e' in the forest that need to be swapped with them). This step can be done efficiently in an amortized sense by finding for each new arrived edge (x, y) the corresponding unique path in the existing forest connecting x and y (if any) and checking the minimum-weight edge on this path [1]. For the updated forest edges, update the last time they were active to the maximum t in the buffer.
  3) ⟨*Output*⟩: Output those edges in the highest marked output buffers and use a *least-recently-used* buffering strategy to re-claim buffer space when necessary.

Data	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert E_t\rvert$
Enron	666	1,297	3,515
LBNL	3,317	9,637	9,258,309
RMBT	101	2815	102,563
TWEET	676,046	2,164,959	3,827,560

Table I
COMMUNICATION GRAPHS USED IN OUR EXPERIMENTS. $|V|$, $|E|$, AND $|E_{ts}|$ ARE THE NUMBER OF VERTICES, EDGES, AND TIME-STAMPED EDGES (I.E. COMMUNICATIONS), RESPECTIVELY.

Complexity: The aforementioned algorithm has runtime complexity $O(\alpha(m_t, n_t) \times m_t)$ where $m_t = |E_t|$, $n_t = |V_t|$, and $\alpha(m_t, n_t)$ is the classical functional inverse of Ackermann's function. So, our algorithm is almost linear in the number of time-stamped edges. Its storage complexity is linear in the number of vertices: $O(k \times n_t)$ [7].

IV. EXPERIMENTS

This section is divided into three subsections: data sets, experimental setup, and results.

A. Data Sets

Table I provides a summary of the communication networks used in our experiments. Enron is the DOJ-released email network between Enron employees from 1999 to 2002.[4] RMBT is the MIT Reality Mining blue-tooth connections collected over 12 months.[5] LBNL is IP traffic collected on an internal enterprise network during a busy hour on 2004.12.15 on port #3 (TCP and UDP compression processes).[6] LBNL data includes scanning activities. TWEET is Twitter data with tweets from 2006 to 2009.[7]

B. Experimental Setup

Our approach produces metrics on graph elements (i.e., vertices, edges, subgraphs) of a time-varying network. Table II lists some of these metrics.

In our experiments, we analyze (a) how these metric relate to each other, (b) how they vary over time, and (c) how they can be used to detect novel discrepancies. Our experimental setup is as follows. For each time t of the system (i.e., when a graph element becomes active), we construct a graph $G(t)$ and compute the metrics listed in Table II for the novel edges (as defined in Section III-B). Since the number of (active) times can be very large, our experiments only consider those times when the system acceleration is at a local maxima or

[4] http://www.cs.cmu.edu/∼enron/

[5] http://reality.media.mit.edu/

[6] http://www.icir.org/enterprise-tracing/download.html

[7] http://www.public.asu.edu/∼mdechoud/datasets.html

Metric	Definition
$Frequency(z,t) = \lvert T_{e,t} \rvert$	Number of times the graph element z has occurred up to time-stamp t.
$FiringRate(z,t) = fr(z,t)$	$\frac{1}{t} Frequency(z,t)$, where t is the number of times that the system has been active.
$Acceleration(z,t) = acc(z,t)$	$\frac{1}{\Delta(t)}(FiringRate(z,t) - FiringRate(z,t-\Delta(t)))$, where $\Delta(t)$ = time difference between the current and last occurrence of z.
$\chi(z,t)$	Starts at 0; increments by 1 if $FiringRate(z,t)$ agrees with the system's firing rate; decrements by 1 otherwise.
$Acceleration\chi(z,t)$	Starts at 0; increments by 1 if $Acceleration(z,t)$ agrees with the system's acceleration; decrements by 1 otherwise.
$SumOf\chi(v,t)$	Sum of χ's for vertex v's incident edges up to t.
$SBP(t)$	Theoretical discrepancy maximum of the set-system at t.
$SbpVertexWeight(v,t) = \chi\_wgt(v,t)$	$\lvert\lvert SumOf\chi(v,t)\rvert - SBP(t)\rvert$
$SbpEdgeWeight(e,t) = \chi\_wgt(e,t)$	$\lvert\lvert SumOf\chi(e,t)\rvert - SBP(t)\rvert$
$SumOfAcceleration\chi(v,t)$	Sum of acceleration χ values for vertex v's incident edges at t.

Table II
Some of the metrics generated by our DND algorithm. A graph element can be a vertex, an edge, or a subgraph. The "system" refers to the communication network as a whole.

minima. For brevity, we mostly report on metrics over vertices.

C. Results

Table III summaries the various experiments reported here.

Figure 2 presents the pairwise (Pearson) correlation coefficients of our various metrics (as described in Table II). For each data set, an entry (I, J) represents the correlation coefficient for metrics I and J across all times and all vertices:

$$\begin{aligned} \rho(I,J) &= \frac{cov(I,J)}{(\sigma_I \times \sigma_J)} \\ I &= \{\forall v \in V \& \forall t \in [t_{min}, t_{max}] : i_{v,t}\} \\ J &= \{\forall v \in V \& \forall t \in [t_{min}, t_{max}] : j_{v,t}\} \end{aligned}$$

As expected, the correlation values differ across various communication networks. This is mainly due to the different types of communications (emails, bluetooth connections, IP traffic, tweets) in our data sets. Here are some noteworthy observations:

- Frequency and firing rate are highly correlated across different types of communications ($\rho \geq 0.8$). Recall that firing rate measures the velocity of a graph element and is a time-normalized measure on frequency.
- Firing rate and χ are highly correlated in Enron ($\rho = 0.76$), less correlated in RMBT ($\rho = 0.55$), and highly uncorrelated in LBNL and TWEET ($\rho = -0.72$). Intuitively, this observation states that in Enron there is agreement between an individual and the system's communication patterns–highlighting the homogenous nature of email communications between Enron employees. In LBNL and TWEET, such agreements disappear, which indicates the heterogenous nature of communications in IP traffic and tweets.
- χ and $SumOf\chi$ are highly correlated in the Enron, LBNL, and TWEET ($\rho > 0.76$), where there is synchronicity in the system's communications; but this is not the case in RMBT's bluetooth connections ($\rho = 0.13$). *Synchronicity* here means that a vertex simultaneously communicates with other vertices (i.e., its outgoing edges occur around the same time). Intuitively, this observation makes sense since bluetooth connections are less intentional (and hence more random) than emails, IP traffic, or tweets.
- $SumOf\chi$ and SBP vertex weight are uncorrelated ($\rho \leq -0.55$) in all but the Enron data set (where $\rho = 0.60$). This is because the ratio of Enron employee's with negative χ values is small (24.2%). In other words, the activities of the majority of Enron vertices (75.8%) agree with the system's activity. In contrast, 54.5% of the RMBT vertices and 100% of the LBNL vertices have negative χ values.
- $Acceleration\chi$ and $SumOfAcceleration\chi$'s are highly correlated in RMBT and LBNL ($\rho > 0.81$) because the majority of the vertices have negative χ values w.r.t. acceleration (i.e., their acceleration behavior disagrees with the system's). In LBNL, 73.6% of the vertices have negative $Acceleration\chi$

Figure	Description
2	Pearson correlation coefficient on DND's various discrepancy-based metrics
3	Scatter-plots of DND's various discrepancy-based metrics across *all* times
4	Scatter-plots of DND's various discrepancy-based metrics at the *last* time-stamp
5	Detecting novel edges with DND's discrepancy-based metrics
6	Some Enron employees highlighted by our DND algorithm

Table III
SUMMARY OF OUR EXPERIMENTS.

Enron	Frequency	Firing Rate	χ	Acceleration χ	Acceleration	Sum of χ's	Sum of Acc. X's	SBP Vertex Wgt
Frequency	1.00	0.87	0.90	0.72	0.01	0.80	0.65	0.46
Firing Rate		1.00	0.76	0.71	0.07	0.65	0.63	0.27
χ			1.00	0.43	0.03	0.97	0.63	0.60
Acceleration χ				1.00	-0.01	0.23	0.41	0.14
Acceleration					1.00	0.03	0.02	0.02
Sum of χ's						1.00	0.60	0.60
Sum of Acc. X's							1.00	0.26
SBP Vertex Wgt								1.00

RMBT	Frequency	Firing Rate	χ	Acceleration χ	Acceleration	Sum of χ's	Sum of Acc. X's	SBP Vertex Wgt
Frequency	1.00	0.80	0.65	-0.89	0.00	-0.51	-0.85	0.67
Firing Rate		1.00	0.55	-0.72	0.07	-0.47	-0.67	0.60
χ			1.00	-0.66	0.00	0.13	-0.62	0.25
Acceleration χ				1.00	-0.01	0.48	0.81	-0.63
Acceleration					1.00	-0.01	-0.01	0.00
Sum of χ's						1.00	0.38	-0.71
Sum of Acc. X's							1.00	-0.54
SBP Vertex Wgt								1.00

LBNL	Frequency	Firing Rate	χ	Acceleration χ	Acceleration	Sum of χ's	Sum of Acc. X's	SBP Vertex Wgt
Frequency	1.00	0.88	-0.87	-0.99	-0.01	-0.99	-0.99	0.99
Firing Rate		1.00	-0.72	-0.88	0.04	-0.86	-0.87	0.85
χ			1.00	0.84	0.01	0.89	0.85	-0.89
Acceleration χ				1.00	0.02	0.98	1.00	-0.98
Acceleration					1.00	0.01	0.02	-0.01
Sum of χ's						1.00	0.97	-1.00
Sum of Acc. X's							1.00	-0.97
SBP Vertex Wgt								1.00

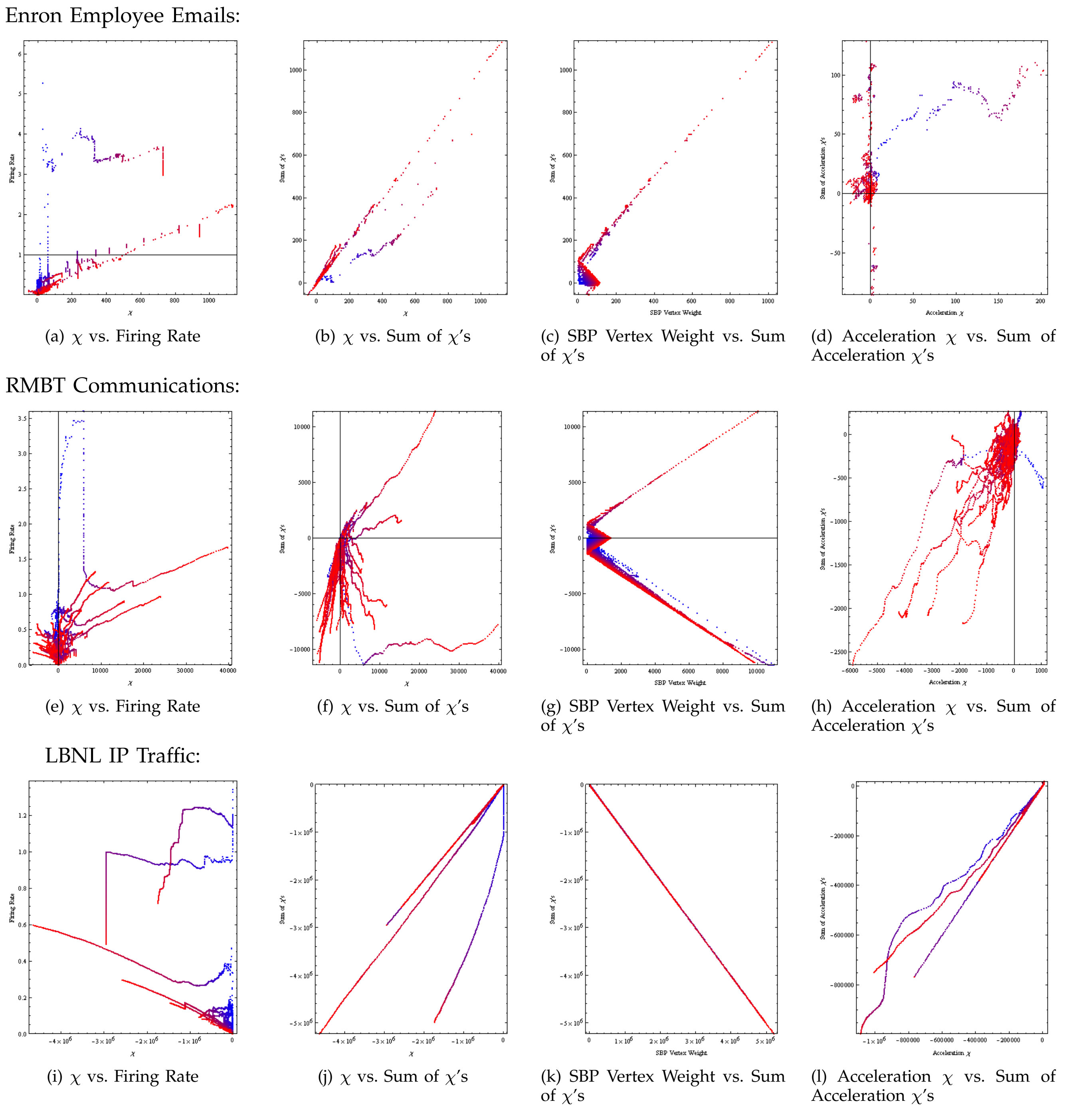
TWEET	Frequency	Firing Rate	χ	Acceleration χ	Acceleration	Sum of χ's	Sum of Acc. X's	SBP Vertex Wgt
Frequency	1.00	1.00	-0.72	-0.59	-0.13	-0.94	-0.11	0.51
Firing Rate		1.00	-0.72	-0.59	-0.13	-0.94	-0.11	0.51
χ			1.00	0.20	0.01	0.76	0.02	-0.21
Acceleration χ				1.00	0.18	0.52	0.36	-0.40
Acceleration					1.00	0.08	-0.02	-0.15
Sum of χ's						1.00	0.03	-0.55
Sum of Acc. X's							1.00	-0.02
SBP Vertex Wgt								1.00

Figure 2. Pearson correlation between various outputted metrics. The colors indicate correlation levels and go from dark green (highly correlated) to dark red (highly uncorrelated).

values; in RMBT, the number is 65%; and in Enron, the number is a mere 23.6%.

Figure 3 depicts the scatter-plots of various metrics on our data sets. Our observations are as follows:

- χ vs. *Firing Rate*: These plots show whether there was agreement between individual vertices and the system (i.e., network) as a whole. The vertices that lay on the $\chi \cong$ *firing fate* line agree with the system. The vertices that have negative χ values disagree with the system.



(a) χ vs. Firing Rate

(b) χ vs. Sum of χ's

(c) SBP Vertex Weight vs. Sum of χ's

(d) Acceleration χ vs. Sum of Acceleration χ's

(e) χ vs. Firing Rate

(f) χ vs. Sum of χ's

(g) SBP Vertex Weight vs. Sum of χ's

(h) Acceleration χ vs. Sum of Acceleration χ's

(i) χ vs. Firing Rate

(j) χ vs. Sum of χ's

(k) SBP Vertex Weight vs. Sum of χ's

(l) Acceleration χ vs. Sum of Acceleration χ's

Figure 3. Scatter-plots of various outputted metrics for Enron, RMBT, and LBNL (best viewed in color). Per plot, each dot represents a vertex. The color of the dot represents time and goes from blue (t_{min}) to red (t_{max}). See text for detailed description.

- χ vs. $SumOf\chi$: These plots show whether the communications in the system were synchronous or asynchronous. The vertices that lay on the $\chi \cong SumOf\chi$ line have synchronized communications. The rest have asynchronous communications. Specifically, when the majority of a vertex' outgoing edges happen around the same time, the behavior of that vertex will be very similar to that of its edges. Hence there will be high correlation between χ and $SumOf\chi$ values. If there are not many outgoing edges from a vertex that happen simultaneously, then the behavior of the vertex as an entity will be quite different from that of its edges (i.e., communications).

- *SBP Vertex Weight* vs. $SumOf\chi$: These plots reflect the significance of χ values. When χ values are positive (as in Enron), the $SumOf\chi$ for a vertex is positively correlated with SBP vertex weight (recall that SBP denotes the theoretical maximum discrepancy). However, when χ values are negative (as in RMBT and LBNL), the $Sumof\chi$ for a vertex is negatively correlated with SBP vertex weight.
- $Acceleration\chi$ vs. $SumOfAcceleration\chi$: These plots showcase the relationship between a vertex' acceleration χ and the sum of acceleration χ values for its incident edges. As the plots show, in RMBT and LBNL this relationship is highly correlated. That is, the acceleration χ of a vertex decreases along with its incident edges. However, such a relationship does not exist in Enron, where a person's acceleration χ can remain the same while its incident edges' acceleration χ change. This observations makes sense for emails where communications between particular pairs of people can accelerate but the communications of each individual stays virtually constant.

Figure 4 shows the scatter-plots for TWEET during the system's last time-stamp. Figure 4(a), depicting χ vs. firing rate for TWEET's last time-stamp, shows that most of the vertices (86.8%) disagree with the system (i.e., have $\chi < 0$). When observed over time, the movement of TWEET's vertices w.r.t. χ and firing rate is noticeably different than the other data sets.[8] Initially, there are vertices that have high firing rates and χ values around zero. Then there is a phase-shift, where the firing rates for these vertices suddenly decreases and other vertices start to appear. These new vertices maintain relatively lower firing rates (because they appear late in the communication stream), but they have high negative χ values (indicating disagreement with the system). Figure 4(b), illustrating χ vs. $SumOf\chi$ for TWEET's last time-stamp, depicts that only 11.4% of vertices have positive values for both χ and $SumOf\chi$. This implies that only 11.4% of the vertices have behaviors that agree with the system. This result indicates that very few edges are active in each time-stamp. Figure 4(c) shows SBP vertex weight vs. $SumOf\chi$ for TWEET's last time-stamp. It illustrates a very regular pattern in the vertices, namely that their movements disagree with the system's (which is not surprising for tweets). Figure 4(d), depicting $Acceleration\chi$ vs. $SumOfAcceleration\chi$ for TWEET's last time-stamp, shows that there are no regular patterns of movement among the acceleration of vertices. Again, this is not surprising given the chaotic and heterogenous nature of communications on Twitter.

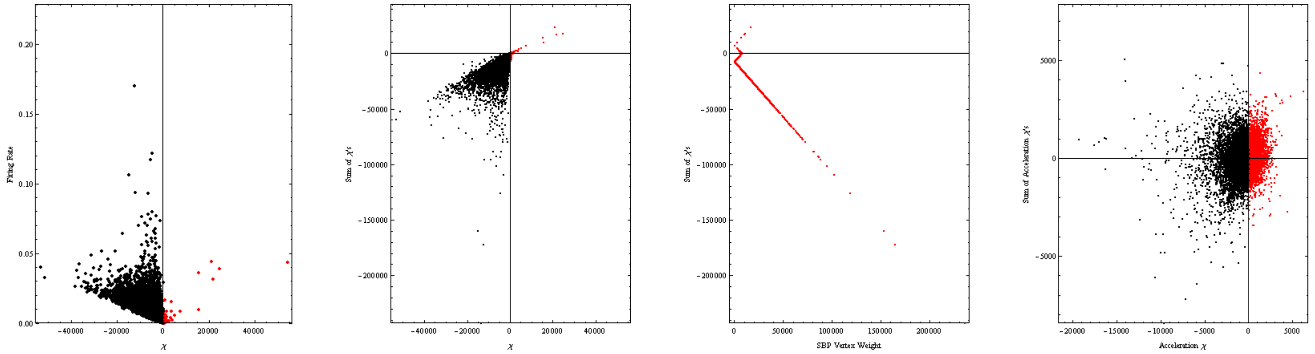
Figure 5 depicts $\chi(e,t)$ and $SBP(t)$ values for Enron's and RMBT's edges. The black curve corresponds to the SBP-bound on edges as a function of time. It represents the theoretical maximum set-system discrepancy. The remaining points in the scatter plot correspond to edges that at a particular time t have a particular $\chi(e,t)$ value. Our DND algorithm looks for those edges that reside i standard-deviations around the $SBP(t)$, the black curve. These edges correspond to our *i-novel* edges. The further from the mean (i.e., higher values of i) tend to produce the "most" novel edges. The same analysis holds for vertices.

Figure 6 highlights the χ vs. firing rate scatter-plots for several Enron employees who stood out in our analysis. First, we observed that a certain set of vertices show similar patterns in their movements over the same period of time. Interestingly these vertices represent people who have close working-relationship between them, such as Linda Robertson (Enron's Chief Lobbyist) and John Shelk (Enron's Vice President for Governmental Affairs). Another observations is that of J. Kaminski. He had a negative χ implying that his activity did not agree with that of the system. J. Kaminski was a Risk Management Expert at Enron, who warned Enron's top executives about the impending dangers.

Recommendations: Based on our results, we recommend the following analysis:

- To detect agreement in communication behavior between a graph element z and the overall network at time t, inspect $\chi(z,t)$ vs. firing rate. Of particular interest are graph elements with negative χ values since they highlight disagreements in behavior.
- To detect synchronicity in the system's communication at time t, examine $\chi(z,t)$ vs. $Sumof\chi(z,t)$ and $Acceleration\chi(z,t)$ vs. $SumofAcceleration\chi(z,t)$. Of particular interest are vertices whose communications are not synchronized with their incident edges (i.e., where $\chi(z,t)$ vs. $Sumof\chi(z,t)$ or $Acceleration\chi(z,t)$ vs. $SumofAcceleration\chi(z,t)$ are uncorrelated).
- To detect phase-shifts in communication behavior, track the discrepancy-based metrics from one time-stamp to the next. Of particular interest are elements who have sudden changes in their discrepancy-based metrics.
- To detect novel graph elements at time t, examine the graph elements whose $\chi(z,t)$ values are i

[8] We intend to release videos of that depict this. We have omitted them here because of the double-blind review rules.



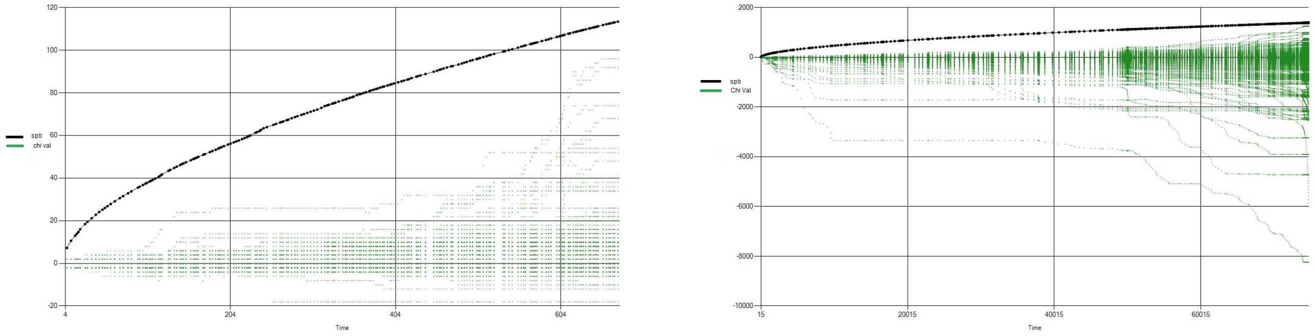
(a) χ vs. Firing Rate

(b) χ vs. Sum of χ's

(c) SBP Vertex Weight vs. Sum of χ's

(d) Acceleration χ vs. Sum of Acceleration χ's

Figure 4. TWEET last time-stamp: scatter plots of various outputted metrics. Per plot, a red point indicates increased activity for a vertex since its last occurrence; a black point indicates the opposite (i.e., no increased activity since the vertex' last occurrence).



(a) Enron: $\chi(e,t)$ (in green) over time & SBP(t) (in black) over time

(b) RMBT: $\chi(e,t)$ (in green) over time & SBP(t) (in black) over time

Figure 5. Enron and RMBT: χ values on edges (green points) and SBP values on edges (black points) over time (x-axis). At time t, an edge is considered *i-novel* if its χ value is i standard deviations away from the SBP value.

standard-deviations from $SBP(t)$.

V. Conclusions

We have introduced combinatorial set-system discrepancy as a useful mathematical construct that is able to isolate characteristic patterns in time-evolving (particularly, communication) networks. Our initial experimentation with statistics based on firing rate and acceleration demonstrates how the corresponding discrepancies can be used to isolate novel patterns (such as synchronicity among communications) in a variety of data sets (Enron emails, LBNL IP traffic, Reality Mining blue-tooth connections, and Twitter tweets). Our approach can be used to build communication-pattern profiles, which can then be re-used in subsequent analysis.

Future work includes investigating how to sample from the activity sequence of a communication network a time sub-sequence such that its associated discrepancy computations produce good approximations to the discrepancy computations over the entire network. We hope this work entices other researchers to use combinatorial discrepancy as a principled approach to advance our understanding of the evolution of large communication networks.

VI. Acknowledgments

This work was performed under the auspices of DyDAN (a DIMACS Center of Excellence), CCICADA (a U.S. Department of Homeland Security Center of Excellence), and the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract No. DE-AC52-07NA27344.

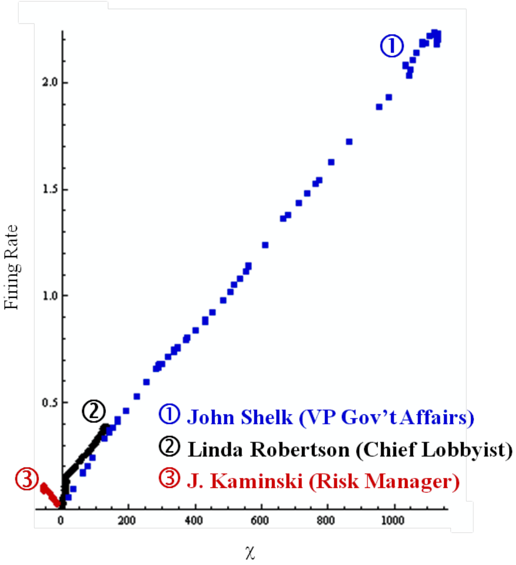


Figure 6. Enron employees who stand out based on their values for χ and firing rate. Plots #1 and #2 (in $\chi > 0$) correspond to Enron lobbyists, who show similar patterns in their emails over the same period of time. Plot #3 (in $\chi < 0$) represents J. Kaminski, a risk management expert who notified Enron top executives about the potential economic bust.

References

[1] J. Abello, A. L. Buchsbaum, and J. Westbrook. A functional approach to external graph algorithms. *Algorithmica*, 32(3):437–458, 2002.

[2] C. Aggarwal and S. Yu. An effective and efficient algorithm for high-dimensional outlier detection. *The VLDB Journal*, 14(2):211–221, 2005.

[3] L. Akoglu, M. McGlohon, and C. Faloutsos. OddBall: Spotting anomalies in weighted graphs. In *PAKDD*, pages 410–421, 2010.

[4] K. M. Borgwardt, H.-P. Kriegel, and P. Wackersreuther. Pattern mining in frequent dynamic subgraphs. In *ICDM*, pages 818–822, 2006.

[5] D. Chakrabarti. Autopart: Parameter-free graph partitioning and outlier detection. In *PKDD*, pages 112–124, 2004.

[6] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3), 2009.

[7] B. Chazelle. An mst algorithm with inverse ackerman's time complexity. *Journal of the ACM*, 47(6):1028–1047, 2000.

[8] B. Chazelle. *The Discrepancy Method*. Cambridge University Press, 2001.

[9] W. Eberle and L. Holder. Anomaly detection in data represented as graphs. *Intell. Data Anal.*, 11(6):663–689, 2007.

[10] W. Eberle and L. B. Holder. Mining for structural anomalies in graph-based data. In *DMIN*, pages 376–389, 2007.

[11] R. T. H. Kaplan, N. Shafrir. Union-find with deletions. In *SODA*, pages 19–28, 2002.

[12] S. Hirose, K. Yamanishi, T. Nakata, and R. Fujimaki. Network anomaly detection based on eigen equation compression. In *KDD*, pages 1185–1194, 2009.

[13] T. Idé and H. Kashima. Eigenspace-based anomaly detection in computer systems. In *KDD*, pages 440–449, 2004.

[14] J.-G. Lee, J. Han, and X. Li. Trajectory outlier detection: A partition-and-detect framework. In *ICDE*, pages 140–149, 2008.

[15] C. Liu, X. Yan, H. Yu, J. Han, and P. S. Yu. Mining behavior graphs for "backtrace" of noncrashing bugs. In *SDM*, 2005.

[16] C. C. Noble and D. J. Cook. Graph-based anomaly detection. In *KDD*, pages 631–636, 2003.

[17] B. A. Prakash, N. Valler, D. Andersen, M. Faloutsos, and C. Faloutsos. BGP-lens: Patterns and anomalies in internet routing updates. In *KDD*, pages 1315–1324, 2009.

[18] J. Sun, S. Papadimitriou, P. S. Yu, and C. Faloutsos. Graphscope: Parameter-free mining of large time-evolving graphs. In *KDD*, pages 687–696, 2007.

[19] B. Thompson and T. Eliassi-Rad. DAPA-V10: Discovery and analysis of patterns and anomalies in volatile time-evolving networks. In *Notes of the 1st Workshop on Information in Networks (WIN)*, September 2009.

[20] B. Thompson and T. Eliassi-Rad. A renewal theory approach to anomaly detection in communication networks. In *Notes of the 2nd Workshop on Information in Networks (WIN)*, September 2010.

[21] B. Wackersreuther, P. Wackersreuther, A. Oswald, C. Böhm, and K. Borgwardt. Frequent subgraph discovery in dynamic networks. In *Notes of the 8th Workshop on Mining and Learning with Graphs (MLG)*, 2010.