Using Partial Probes to Infer Network States

Pavan Rangudu* °, Bijaya Adhikari*, B. Aditya Prakash*, Anil Vullikanti* °
*Department of Computer Science, Virginia Tech
°NDSSL, Biocomplexity Institute, Virginia Tech
rangudu@vt.edu, {bijaya, badityap}@cs.vt.edu, vsakumar@vt.edu

ABSTRACT

In many applications, such as the Internet and infrastructure networks, nodes fail or get congested dynamically. We study the problem of inferring all the failed nodes, when only a sample of the failures is known, and there exist correlations between node failures/congestion in networks. We formalize this as the GraphStateInf problem, using the the Minimum Description Length (MDL) principle. We propose a greedy algorithm for minimizing the MDL cost, and show that it gives an additive approximation, relative to the optimal. We evaluate our methods on synthetic and real datasets, which includes one from WAZE which gives traffic incident reports for the city of Boston. We find that our method gives promising results in recovering the missing failures.

1 INTRODUCTION

Most network applications assume the network is static, and is known ahead of time. This is not true in practice, and networks are inferred by indirect measurements, e.g., as in the case of the Internet router/AS level graphs, which are constructed using traceroutes, e.g., [5], or biological networks, which are inferred by experimental correlations, e.g., [16]. Further, network elements can fail dynamically, or their state may change with time. For instance, links in the Internet router network or the transportation network can get congested or fail. Reconstructing the network topology dynamically and inferring network states in such settings are fundamental problems. Such problems have been studied as part of the area of "network tomography", especially in communication networks, e.g., [9, 11, 19]. Such networks are not publicly accessible, and indirect probes are the only means of obtaining information; examples of probes include queries of the activity states of selected nodes and end-to-end measurements of delays between selected pairs of nodes. These become very challenging problems, and all prior work in this direction in network tomography has been focused on simple models of *independent* link failures and delays, e.g., with exponentially distributed probabilities [11].

In many settings, such as disaster events in infrastructure networks, however, failures might be spatially correlated, as in [1, 3, 15]. For instance, in the model considered in [1], the probability that a node j fails decays with the distance from a source s. This motivates the problem GraphStateInf of inferring the network states under such spatial correlations, which is the focus of our paper. A closely related topic is the inference of the source of an infection and other missing infections in the case of epidemic spread on networks—these are typically modeled as SI/SIR processes, where the infection spreads from one node to its neighbors with some probability [12, 13, 17, 18]; see [8, 10] for introduction to epidemic models. An approach based on the Minimum Description Length (MDL) principle [7, 14] was introduced for missing infection problems in [12, 13, 18] for the SI process. However, there are key differences in our setting, and the methods developed for missing node infection in epidemic processes do not seem to directly work for the GraphStateInf problem. Our contributions are summarized below.

(1) We develop a novel formulation for the GraphStateInf problem using the Minimum Description Length (MDL) principle, which takes correlated failures into account. We present Greedy, an algorithm for inferring the missing failed nodes, given a sample of the failures. We prove that the MDL cost of the solution computed by Greedy is within an additive approximation of the minimum cost MDL solution. Typically, approaches using MDL are based on heuristics and getting bounds is non trivial as MDL cost functions are not convex. To the best of our knowledge, our algorithm is the first to obtain rigorous bounds on the objective value among MDL based approaches for network inference.
(2) We evaluate our results on different kinds of synthetic and real datasets, namely, one week's worth of traffic status and incident reports from WAZE for the city of Boston, and electric disturbance events in the power grid (described in Section 4.1). We study the precision, recall and F1-score for Greedy, compared with a baseline. We observe that our algorithm is quite effective in inferring unknown/missing failures in the network, and has lower MDL cost than the baseline.

The rest of the paper is organized in the usual way, with the formulation first, then methods, and experiments. We then finally give the related work and conclusions.

2 OUR PROBLEM FORMULATION

We are given an undirected graph $G(V, E)$ representing an infrastructure network. We assume there is an initial failure at a node, referred to as the seed node, which causes other nodes to fail. Further, a subset $Q \subseteq I$ of the actual failed nodes are assumed to be known. The objective in the GraphStateInf problem is to infer all the missing failures.

2.1 Failure Model

Next, we will discuss the failure model we use to describe the failures in the given network G. This model is motivated by the geographically correlated failure model introduced by Agarwal et al [1], to capture failures in infrastructure networks due to large scale disasters. In such events, there is an initial localized failure, which causes other nodes to fail with some probability that decays with the distance from the source.

Following [1], we assume there is an initial single 'seed' node s and all the failures I in G are caused due to the influence of that seed node. We assume a discrete probability distribution function $p_s : V \to [0, 1]$ that gives the probability of each node $v \in V$ being a seed and conditional failure probability distribution function $F : V \times V \to [0, 1]$ that gives the failure probability of a node $v \in V$ given a seed node s. Note that the $p_s(v)$ is the probability of v being the only seed, i.e, $\sum_{v \in V} p_s(v) = 1$. These probability distributions are precomputed from historically observed failures. We assume that the conditional failure probabilities given by F are independent i.e., for-all $v_1, v_2 \in V$ and $v_1 \neq v_2$,

$$F(v_1 \cap v_2 \mid s) = F(v_1 \mid s)F(v_2 \mid s) \tag{1}$$

2.2 Probes

Based on our model given above, we assume that some seed failed causing multiple correlated failures across the network G. The final set of true failures is represented by $I \subseteq V$. Further, we are also given a set of failed nodes represented by $Q \subseteq I$, which we will refer to as probes in rest of this paper. We assume that the given set of input probes Q are sampled uniformly at random from the true failure set I with probability γ.

2.3 MDL

We formulate our problem using the Minimum Description Length (MDL) principle [6]. We will use two-part MDL, or the sender-receiver framework. Our goal here is to transmit the given set of probes Q from sender to receiver by assuming that both of them know the layout of the network G. We do this by identifying the model that best describes the given data in terms of a formal objective or cost function. This cost function consists of two parts:

(1) Model cost that signifies the complexity of the selected model that explains the failures in the network; and
(2) Data cost that represents the cost of observing the given probe data Q given the model.

More formally, given a set of models $\mathcal{M}$, MDL identifies the best model M^* as the model that minimizes $\mathcal{L}(M) + \mathcal{L}(\mathcal{D} \mid M)$, in which $\mathcal{L}(M)$ is the model-cost (length in bits to describe model M), and $\mathcal{L}(\mathcal{D}|M)$ is the data-cost (the length in bits to describe the data using M). Note that the data we need to describe in our situation is the probes set Q (and not the true failures set I). Next we describe the model space and the model and data cost, which we will optimize.

2.4 Model Space and Cost

Model Space: The most natural model for our problem would have been $\mathcal{M} = (s, I)$ (the source $s \in V$ and the full failure set $I \subseteq V$), as it directly mimics the generative process of the failure model. However, this model has several disadvantages. Firstly, note that this model space is intuitively 'fragile': small changes in I or the source s can have vastly different costs. Hence due to data sparsity, we expect it would be very hard to learn the true source which generated the failures—indeed, in our experiments, we find that it was not robustly learning the true source. As a result, we also found that the solutions with minimum MDL cost were finding very few missing failed nodes (i.e. $I - Q$), leading to a very low recall. How to design a better model space for our problem? We observe that this model intuitively tries to explain 'more' than what is needed. Note that while our original goal was to map the missing *failures* only, this approach tries to explain the source as well as the set of failures. Hence we adopt a different approach, where we try to marginalize over the seeds, and focus only on the failures. This makes our model space more robust as well. This motivates our proposed model, which consists of three components, namely, $\mathcal{M} = (|Q|, |I|, I)$. In other words, we send the size of probes Q, the size of true failure set I, and then identify the set itself. After sending the model, we will then identify the actual probes set Q as the data.

Model cost: The MDL model cost, $\mathcal{L}(|Q|, |I|, I)$ has three components

$$\mathcal{L}(|Q|, |I|, I) = \mathcal{L}(|Q|) + \mathcal{L}\big(|I| \,\big|\, |Q|\big) + \mathcal{L}\big(I \,\big|\, |Q|, |I|\big).$$

We derive these below. We have $\mathcal{L}(|Q|) = -\log\Big(Pr(|Q|)\Big)$, by using the *Shannon-Fano* code to encode $|Q|$. Similarly we have:

$$\begin{aligned}\mathcal{L}\big(|I| \,\big|\, |Q|\big) &= -\log\Big(Pr\big(|I| \,\big|\, |Q|\big)\Big) \\ &= -\log\Big(\frac{Pr(|Q| \,\big|\, |I|)Pr(|I|)}{Pr(|Q|)}\Big)\end{aligned} \tag{2}$$

From the sampling assumption for Q, we can get:

$$Pr\big(|Q| \,\big|\, |I|\big) = \binom{|I|}{|Q|}\gamma^{|Q|}(1-\gamma)^{|I\setminus Q|} \tag{3}$$

Also observe that:

$$\begin{aligned}\mathcal{L}\big(I \,\big|\, |Q|, |I|\big) &= -\log\Big(Pr\big(I \,\big|\, |Q|, |I|\big)\Big) \\ &= -\log\Big(Pr\big(I \,\big|\, |I|\big)\Big) = -\log\Big(\frac{Pr(I)}{Pr(|I|)}\Big)\end{aligned} \tag{4}$$

Combining all of the above, the complete model cost is:

$$\begin{aligned}\mathcal{L}(|Q|, |I|, I) &= \mathcal{L}(|Q|) + \mathcal{L}\big(|I| \,\big|\, |Q|\big) + \mathcal{L}\big(I \,\big|\, |Q|, |I|\big) \\ &= -\log\Big(Pr(|Q|)\Big) - \log\Big(\frac{Pr(|Q| \,\big|\, |I|)Pr(|I|)}{Pr(|Q|)}\Big) \\ &\quad - \log\Big(\frac{Pr(I)}{Pr(|I|)}\Big) \\ &= -\log\Big(Pr\big(|Q| \,\big|\, |I|\big)\Big) - \log\Big(Pr(|I|)\Big) - \log\Big(\frac{Pr(I)}{Pr(|I|)}\Big) \\ &= -\log\Big(\binom{|I|}{|Q|}\gamma^{|Q|}(1-\gamma)^{|I\setminus Q|}\Big) \\ &\quad - \log\Big(\sum_{s \in V} Pr(I \mid s)p(s)\Big) \\ &= -\log\binom{|I|}{|Q|} - |Q|\log(\gamma) - (|I| - |Q|)\log(1-\gamma) \\ &\quad - \log\Big(\sum_{s \in V} p_s(s)\prod_{v \in I} F(v \mid s)\prod_{v' \notin I}\big(1 - F(v' \mid s)\big)\Big)\end{aligned} \tag{5}$$

2.5 Data Cost

Now, we need to describe the given input probes Q in terms of the model. Given model $\mathcal{M} = (|Q|, |I|, I)$, describing Q is the same as specifying the adjustments that needs to be applied to the failure set I in the model to reach Q, which can be done by describing the following sets:

(1) Unobserved failures i.e., $Q^+ = I \setminus Q$
(2) Observation errors i.e., $Q^- = Q \setminus I$

In this paper, we assume that there are no observation errors, i.e., $Q^- = \emptyset$ (as $Q \subseteq I$). According to the sampling assumption we have, Q is sampled uniformly at random from I with probability γ. This implies that $Q^+ = I \setminus Q$ is sampled from I with uniform probability $(1-\gamma)$. Hence we can compute the probability of seeing a set Q^+ when sampled from I as follows

$$Pr(Q^+|I) = \gamma^{|Q|}(1-\gamma)^{|Q^+|} \tag{6}$$

Now, using this probability distribution of observing the set Q^+ given the failure set I we can compute the optimal number of bits required to transmit Q^+ encoded in terms of model M as follows (again using the Shannon-Fano code):

$$\begin{aligned}\mathcal{L}(Q^+|I) &= -\log\left(\gamma^{|Q|}(1-\gamma)^{|Q^+|}\right)\\ &= -|Q|\log(\gamma) - (|I|-|Q|)\log(1-\gamma)\end{aligned} \tag{7}$$

2.6 Our Formal Problem

Putting it all together, we can state our formal problem:

Given an undirected graph $G(V,E)$, where node failures taken place in the network as per the model described in Section-2.1, and a set of observed failures $Q \subseteq V$, which are sampled independently from the true failure set I^* with a uniform probability γ, find the complete set of failures $I \subseteq V$ by minimizing the MDL cost function $\mathcal{L}(|Q|, |I|, I, Q)$ given by

$$\begin{aligned}\mathcal{L}(|Q|,\,|I|,\,I,\,Q) &= \mathcal{L}(|Q|)+\mathcal{L}(|I|\,|\,|Q|)\\ &\quad +\mathcal{L}(I\,|\,|Q|,\,|I|) + \mathcal{L}(Q\,|\,|Q|,\,|I|,\,I)\\ &= -\log\binom{|I|}{|Q|} - \log\Big(\sum_{s\in V} p_s(s)\prod_{v\in I}F(v\mid s)\prod_{v'\notin I}\left(1-F(v'\mid s)\right)\Big)\\ &\quad -2|Q|\log(\gamma) - 2(|I|-|Q|)\log(1-\gamma)\end{aligned} \tag{8}$$

where $p_s(s)$ is the seed probability of s and $F(v\mid s)$ is the failure probability of node v given seed node s.

3 PROPOSED METHODS

Clearly the search space for the problem is large, and there exists no trivial structure for fast search. We now describe two approaches for finding solutions with low MDL cost. The first, LocalSearch, incrementally adds a node that gives the most reduction in MDL cost, till no further improvements occur. The second, Greedy, guesses the size k of the optimal solution, and greedily picks the k nodes that would minimize the cost. We show that the cost of the solution produced by Greedy is within an additive factor of the optimum.

3.1 Algorithm LocalSearch

This is an intuitive local search algorithm which is popularly used in many MDL optimzations. We initialize $\hat{I}$ to Q, and just repeatedly add a node that reduces the MDL cost. This is described formally in Algorithm 1.

3.2 Algorithm Greedy

In this section we will discuss an efficient algorithm for finding a failure set I which provides an additive approximation guarantee on the MDL cost of the solution.

Algorithm 1 Algorithm LocalSearch

Input: Instance (V, Q, p, P, γ)
Output: Solution $\hat{I}$ that minimizes $\mathcal{L}(|Q|, |\hat{I}|, \hat{I}, Q)$

$\hat{I} \leftarrow Q$
while $\exists v \in V\setminus\hat{I} : \mathcal{L}(|Q|,|\hat{I}|,\hat{I},Q) - \mathcal{L}(|Q|,|\hat{I}|+1,\hat{I}\cup\{v\},Q) > 0$ **do**
  $u \leftarrow \arg\max_{v\in V\setminus\hat{I}} \mathcal{L}(|Q|,|\hat{I}|,\hat{I},Q) - \mathcal{L}(|Q|,|\hat{I}|+1, I\cup\{v\}, Q)$
  $\hat{I} \leftarrow \hat{I}\cup\{u\}$
end while
Return $\hat{I}$

First, let $A = -\log\binom{|I|}{|Q|} - 2|Q|\log(\gamma)$. We rewrite the MDL cost function in the following manner:

$$\begin{aligned}\mathcal{L}(|Q|,|I|,I,Q) =& A - \log\Big(\sum_{s\in V} p_s(s)\prod_{v\in I}F(v\mid s)\prod_{v'\notin I}\left(1-F(v'\mid s)\right)\Big)\\ &- 2(|I|-|Q|)\log(1-\gamma)\\ =& A - \log\Big(\sum_{s\in V} p_s(s)\prod_{v\in V}\left(1-F(v'\mid s)\right)\\ &\prod_{v\in I}\frac{F(v\mid s)}{(1-F(v\mid s))}\Big) - \log(1-\gamma)^{2(|I|-|Q|)}\\ =& A - \log\Big(\sum_{s\in V} p_s(s)\prod_{v\in V}\left(1-F(v'\mid s)\right)(1-\gamma)^{-2|Q|}\\ &\prod_{v\in I}\frac{F(v\mid s)(1-\gamma)^{2|I|}}{(1-F(v\mid s))}\Big)\\ =& A - \log\Big(\sum_{s\in V} g(s)\prod_{v\in I} f(s,v)\Big),\end{aligned} \tag{9}$$

where $g(s) = (1-\gamma)^{-2|Q|}p_s(s)\prod_{v\in V}\left(1-F(v\mid s)\right)$ and $f(s,v) = \frac{F(v\mid s)(1-\gamma)^2}{1-F(v\mid s)}$. Therefore, the problem reduces to finding a set $\hat{I}$ such that

$$\begin{aligned}\hat{I} = \arg\min_I \Big\{ &-\log\binom{|I|}{|Q|} + 2|Q|\lambda_1\\ &-\log\Big(\sum_{s\in V} g(s)\prod_{v\in I} f(s,v)\Big)\Big\}.\end{aligned} \tag{10}$$

The main idea of this algorithm is to use the quantity $f(s,v) = \frac{F(v\mid s)(1-\gamma)^2}{1-F(v\mid s)}$ defined above as the 'weight' for each pair (s,v). For each seed node $s \in V$, we guess the size of the solution $|I_s|$, if the source were to be s, and pick the set of $|I_s|$ nodes which minimizes the MDL cost. This is described formally in Algorithm 2, and analyzed in formally Theorem 3.1.

Theorem 3.1. *Let I^* be the set minimizing the MDL cost, and let I denote the solution computed by Algorithm Greedy. Then, $\mathcal{L}(|Q|,|I|,I,Q) \le \mathcal{L}(|Q|,|I^*|,I^*,Q) + \log(n)$, where n is the number of seed nodes.*

Proof. Recall the definitions of $g(s)$, $f(s,v)$ and A above. Then,

$$\mathcal{L}(|Q|,k,I,Q) = -\log\Big(\sum_{s\in V}\phi(s,I)\Big) + A \quad ,$$

Algorithm 2 Algorithm GREEDY

Input: Instance (V, Q, p, P, γ)
Output: Solution $\hat{I}$ that minimizes $\mathcal{L}(|Q|, |\hat{I}|, \hat{I}, Q)$
1: **for** each $s \in V$ **do**
2: **for** each $k \in [|Q|, |V|]$ **do**
3: $I_s(k) \leftarrow$ Top $k - |Q|$ nodes in $V \setminus Q$ with highest weight $f(s, v)$
4: $I_s(k) \leftarrow I_s(k) \cup Q$
5: **end for**
6: **end for**
7: $\mathcal{S} \leftarrow \{I_s(k) : \forall s \in V \& k \in [|Q|, |V|]\}$
8: $\hat{I} \leftarrow \arg\min_{I \in \mathcal{S}} \mathcal{L}(|Q|, |I|, I, Q)$
9: Return $\hat{I}$

where $\phi(s, I) = g(s) \prod_{v \in I} f(s, v)$. Note that $\phi(s, I)$ is maximized for the set $I_s(k)$ defined in Algorithm GREEDY, since this consists of the set of top $k - |Q|$ nodes in $V \setminus Q$, with respect to the quantity $f(s, v)$, along with all nodes in Q. Therefore, we have

$$\phi(s, I_s(k)) \geq \phi(s, I^*)$$

Adding over all possible seed nodes, we have

$$\sum_{s \in V} \phi(s, I_s(|I^*|)) \geq \sum_{s \in V} \phi(s, I^*)$$

which implies for some seed $\hat{s}$, we have

$$\begin{aligned} \phi(\hat{s}, I_{\hat{s}}(k)) &\geq \frac{1}{n} \sum_{s \in V} \phi(s, I^*) \\ \Rightarrow -\log\Big(\phi(\hat{s}, I_{\hat{s}}(k))\Big) &\leq -\log\Big(\frac{1}{n} \sum_{s \in V} \phi(s, I^*)\Big) \\ &= -\log\Big(\sum_{s \in V} \phi(s, I^*)\Big) + \log(n) \end{aligned}$$

This, in turn, implies

$$\begin{aligned} \mathcal{L}(|Q|, k, I_{\hat{s}}(k), Q) &= -\log\Big(\sum_{s \in V} \phi(s, I_{\hat{s}}(k))\Big) + A \\ &\leq -\log\Big(\phi(\hat{s}, I_{\hat{s}}(k))\Big) + A \\ &\leq -\log\Big(\sum_{s \in V} \phi(s, I^*)\Big) + \log(n) + A \\ &\leq \mathcal{L}(|Q|, k, I^*, Q) + \log(n), \end{aligned}$$

where the first inequality follows because $\phi(s, I) \geq 0 \; \forall \; s, I$, so that $\sum_{s \in V} \phi(s, I_{\hat{s}}(k)) \geq \phi(\hat{s}, I_{\hat{s}}(k))$. Since, the Algorithm 2 searches over all possible solution sizes k, the theorem follows. □

LEMMA 3.2. *Algorithm GREEDY runs in $O(|V|^3)$ time.*

PROOF. The quantities $g(s)$ and $f(s, v)$ defined earlier in (9) can be computed for all s, v in $O(|V|^3)$ time. The algorithm involves two for loops. The inner loop in lines 2-5 runs in $O(|V|)$ time, since it finds a solution $I_s(k)$ for each k. The set $\mathcal{S}$ has size $O(|V|^2)$. Step 8 of the algorithm involves computing $\mathcal{L}(|Q|, |I_s(k)|, I_s(k), Q)$ for each set $I_s(k)$. Done naively, it takes $O(|V|^2)$ time to compute this. However, by keeping the intermediate solutions, we can compute $\mathcal{L}(|Q|, |I_s(k)|, I_s(k), Q)$ for a given s and for all k incrementally in $O(|V|^2)$ time, leading to the time bound in the lemma. □

4 EXPERIMENTS

We evaluate performance of our algorithms on various synthetic and real networks, which are discussed next.

4.1 Datasets

Synthetic Grid Dataset. We created a simple 60×60 grid where each cell is considered as a node in a road network, leading to 3600 nodes. We assumed an uniform seed probability distribution across all nodes. We computed conditional failure probabilities (PlainCF) between pair of nodes (s, v) based on Geographically Correlated Failure (GCF) Model [1] i.e., if s is the seed node then

$$F(v \mid s) = 1 - d(s, v) \tag{11}$$

where $d(s, v)$ is a distance function $d : V \times V \rightarrow [0, 1]$. In our case, $d(\cdot, \cdot)$ is the Manhattan distance between the nodes normalized by the maximum distance. We will refer to this set of conditional failure probabilities as GCF.

Real Datasets. We created three datasets from real world node failure logs in transportation and power-grid networks. We use failure logs in road networks from WAZE alerts data, which is publicly available on the City of Boston's website[1]. These alerts have been reported by users via the crowd sourced application WAZE between Monday 23rd February, 2015 and Sunday 1st March, 2015. The alerts in the dataset are spatially distributed across Boston, Cambridge, and Brook-line regions of Massachusetts. The alerts include different types failures such as traffic jam, extreme weather, accidents, and road closures. Additionally, the latitude and longitude of the affected locations, and start and end time of the alert is also given. From these alerts, we created two datasets based on traffic jam (JAM) and extreme weather (WEATHER).

Similarly, we use a list of Electric disturbance events from Energy.gov[2]—this list includes reported events of electric emergencies and disturbance in power supply from 2002 to 2015. Each event log contains information regarding date and time of the beginning and restoration of the event, geographical areas affected by the event, number of customer affected, and so on. We created POWER-GRID dataset from the log of electric emergencies and disturbances.

Dataset creation. As discussed in Section 2.1, we need to define the seed probability and pair-wise conditional failure probability distributions over all nodes in the network. This is done in the following manner. For WAZE alert data, we have partitioned the complete geographical region occupied by these failures by using a 119×78 grid as shown in Figure 1, where each cell is 0.00166° square and acts as a node in our virtual road network. For the POWER-GRID data, each location referred in the dataset acts as a node.

Seed Probability. Let n_v denote the cell/node v, as discussed above, and let $N = \sum_v n_v$ denote the total number of failures across all nodes. We define the seed probabilities as

$$p_s(v) = \frac{n_v}{N} \tag{12}$$

Conditional Failure Probabilities. We construct a Binary Failure State Time Series (BinTS) for a span of 7 days, by using the temporal information that is available from WAZE alerts data. This time series

[1] https://data.cityofboston.gov/
[2] https://www.oe.netl.doe.gov/

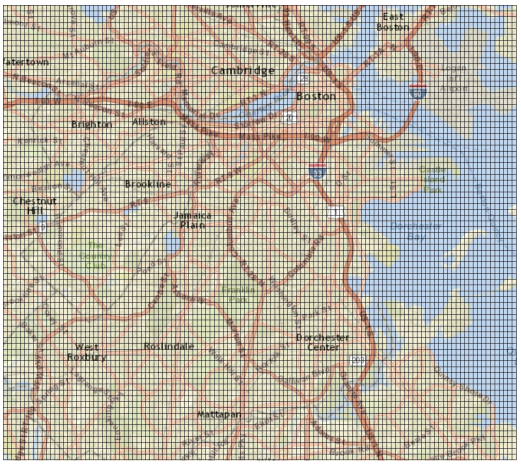


Figure 1: Partitions of Boston region occupied by WAZE alerts using 119×78 grid

gives a binary (0 or 1) value for each time step which represents the failure state of the respective node i.e., $BinTS_v(t) = 1$ implies that there is at-least one failure in v at time t. Using BinTS we were able to compute the pair-wise conditional failure probabilities for our datasets in the following manner. For two nodes v_1 and v_2, we define the Plain Conditional Failure Probability (PlainCF) of v_1 given v_2 as the ratio between number of time steps in which both v_1 and v_2 are failed (i.e. with value 1 in BinTS) to the number of time steps in which only v_2 is failed.

$$PlainCF(v_1 \mid v_2) = \frac{|\{t|\forall t,\, BinTS_{v_1}(t) = 1 \& BinTS_{v_2}(t) = 1\}|}{|\{t|\forall t,\, BinTS_{v_2}(t) = 1\}|} \quad (13)$$

We study two more synthetic conditional failure probabilities, named URandCF and NRandCF, for each dataset; these are defined in the following manner: URandCF is an arbitrary sample from a uniform distribution and NRandCF is an arbitrary sample from a normal distribution ($0.1 \times \mathcal{N}(5, 1)$) over the values $[0, 1]$. We follow the same procedure to generate conditional probability failures for the POWER-GRID dataset.

Dataset descriptions. We construct three different datasets following the above steps.

JAM: This is a dataset that we created from WAZE alerts data using data of failure type JAM. The resultant dataset consists of a road network with 2650 nodes along with seed probability distribution and conditional failure probabilities computed as discussed in Sections 4.1 and 4.1. Figure 2 presents a spatial and frequency distributions of the seed and conditional failure probabilities of this dataset.

WEATHER: Similar to the JAM dataset this dataset is created by using WEATHERHAZARD failure data from WAZE alerts data. The resultant dataset consists of a road network with 1520 nodes along with seed probability distribution and conditional failure probabilities computed as discussed in Sections 4.1 and 4.1.

POWER-GRID: As mentioned earlier, we created this dataset from the log of electrical emergencies and disturbances. We filtered out the events in the log which were not related to loss of electric service. For this dataset, which consists of 24 nodes, we computed failure likelihood and conditional failure probabilities are described in Sections 4.1 and 4.1.

4.2 Performance evaluation

In this section we discuss the performance of our algorithms against various datasets that are described in Section 4.1 across various values of $\gamma \in [0.1, 1.0]$. We examine the precision, recall and F1-score for Greedy, compared with LocalSearch. We observe that our MDL based approach does indeed allow us to infer unknown/missing failures in the network using the probes. The specific MDL formulation we consider in Section 2.6, which includes $|I|$ in the model seems to perform much better than other natural MDL formulations.

Comparison of Greedy with LocalSearch. Figure 3 presents a comparison of the trends in performance of both algorithms on the JAM dataset with PlainCF probabilities across $\gamma \in [0.1, 1.0]$ and MDL cost of their respective solutions. For both algorithms, the performance varies with the sampling rate, γ. We find that the solution computed by Greedy has lower MDL cost, compared to the baseline. One interesting observation from Figure 3b is that the recall for Greedy decays with γ till 0.4 and then increases. Greedy has higher F1-score compared to LocalSearch, for most values of γ. In the rest of our evaluation, we only consider Greedy.

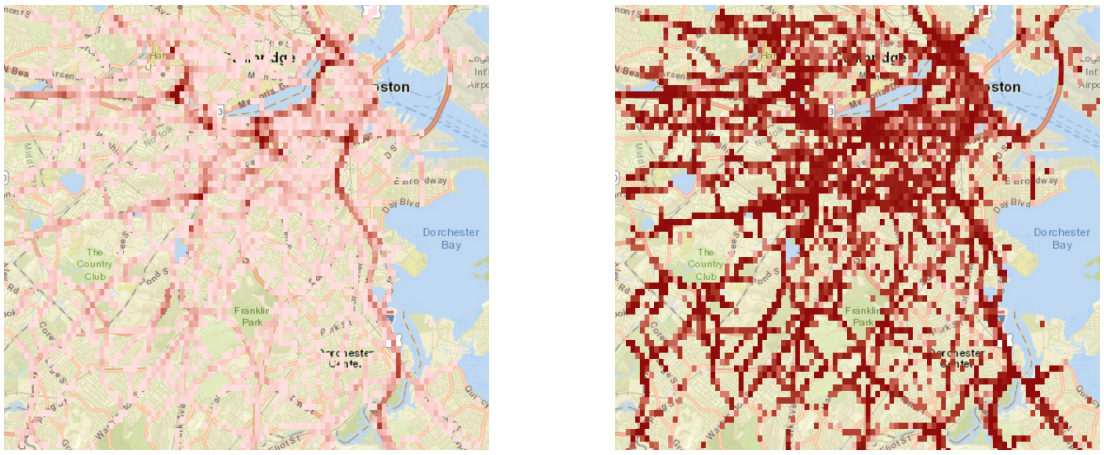
Performance of Greedy for different datasets. Figures 4, 5, 6 and 7 present the performance of Algorithm Greedy for all the datasets from Section 4.1, and for the three different ways of defining conditional probabilities. Across all these results, on average we are able to find 80% of the failed nodes with an average precision of 79% across various values of $\gamma \in [0.1, 1.0]$. In other words, we are successful in inferring a reasonable fraction of unknown/missing failures in the network from partial set of observations with a reasonable precision.

5 RELATED WORK

Some of the different areas related to our work include network and state inference in communication networks, reconstructing networks from cascades and inferring missing infections in the case of epidemics. We briefly discuss these below.

The area of network tomography involves inferring link states, such as delays and failures, in the Internet and other communication networks; see, e.g., [2, 4, 9, 11, 19]. Probes such as end-to-end delays are the only measurements that are available in such networks. At an abstract level, the problem here involves solving for the link delay vector $\mathbf{x}$, given the measured delays across the probes. This becomes a very challenging problem and it is typically assumed that link characteristics such as delays are modeled as *independent* random variables with known distributions, but potentially unknown parameters. Xia et al. [19] solve this assuming link delays are exponentially distributed. Ni et al. [11] study different kinds of probing models, including multicast probes which can give estimates on a tree, and develop methods for inferring the topology in dynamic networks. There has also been work on designing probes to infer part of the network structure, as in [2].

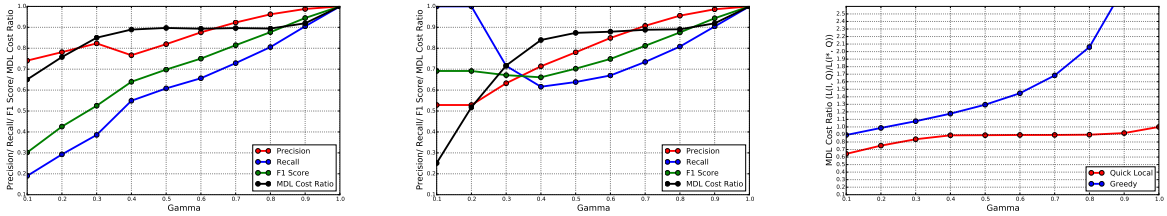
Another related topic is the inference of the source of an infection and other missing infections, in the case of epidemic spread on networks. Epidemics are modeled as stochastic processes, e.g.,



a: Spatial distribution of seed probabilities.

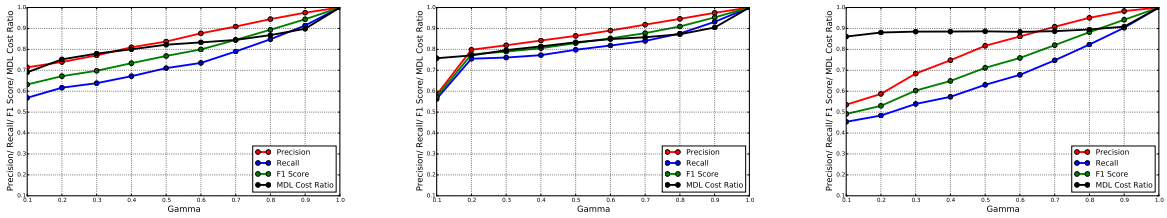
b: Spatial distribution of PlainCF for a random seed node.

Figure 2: JAM dataset created from WAZE alerts data of Boston, Cambridge, Brook-line regions.



a b c

Figure 3: Performance of the LocalSearch (a) and Greedy (b) on the JAM dataset with PlainCF probabilities. The MDL costs of the solutions is shown in (c).

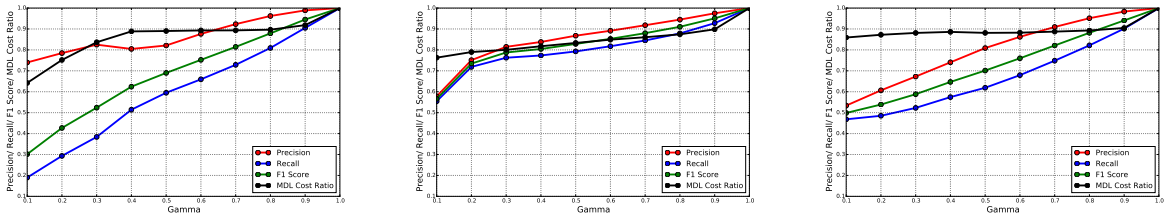


a b c

Figure 4: Performance of Greedy on the Synthetic Grid Dataset with GCF (a), URandCF (b), and NRandCF (c) conditional probabilities.

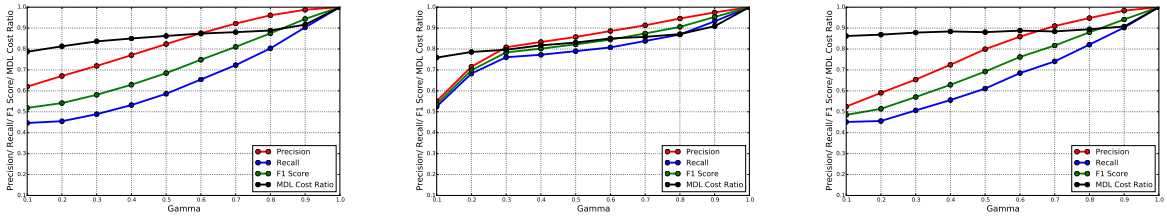
SI/SIR, in which the infection spreads from an infected node to its susceptible neighbors. Usually, only partial information about the infections is known, and some of the problems that have been studied include identifying the source of an infection and finding other missing nodes [12, 13, 17, 18]. The Minimum Description Length (MDL) principle [7, 14] has been successfully used in [12, 13, 18] for these problems, whereas [17] develop an MLE method.

A different class of failure models from the SI/SIR type of epidemics has been studied extensively, motivated by settings such as disaster events, e.g., [1, 3, 15]. These studies assume an initial failure, and subsequent failures whose probability is correlated with the source. For instance, in [1], the probability $p(j|i)$ that node j fails, given that i is the source is a function of the distance from i to j, with the probabilities decaying with the distance. Our work is motivated by these models.



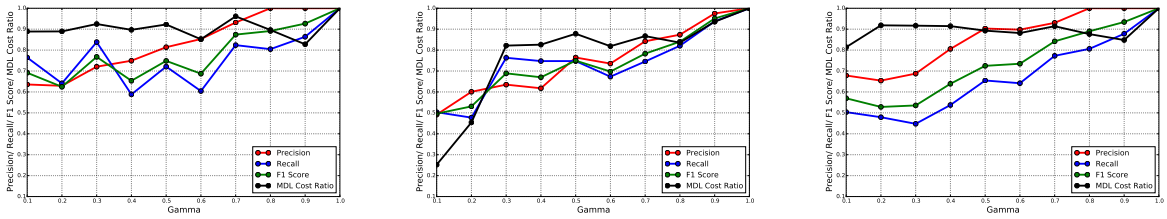
a b c

Figure 5: Performance of Greedy on the JAM Dataset with PlainCF (a), URandCF (b), and NRandCF (c) conditional probabilities.



a b c

Figure 6: Performance of Greedy on the WEATHER Dataset with PlainCF (a), URandCF (b), and NRandCF (c) conditional probabilities.



a b c

Figure 7: Performance of Greedy on the Power Grid Dataset with PlainCF (a), URandCF (b), and NRandCF (c) conditional probabilities.

6 CONCLUSIONS

Our results show that an MDL based approach is quite useful in the problem of inferring missing failures in settings with correlated failures. This motivates its use in other inference problems with partial information. We have considered the simplest notion of a probe here—information about specific nodes which have failed. Extending our work to other kinds of probes (like connectivity queries) is an interesting and natural problem. Inferring the state of the network using such probes, and supporting additional queries are interesting problems. Further we have assumed there are no observational errors—designing robust algorithms in face of errors is also interesting future work.

Acknowledgements

This paper is based on work partially supported by the NSF (IIS-1353346), the NEH (HG-229283- 15), ORNL (Order 4000143330) and from the Maryland Procurement Office (H98230-14-C-0127), a Facebook faculty gift, DTRA CNIMS Contract HDTRA1-11-D-0016-0010, NSF BIG DATA Grant IIS-1633028 and NSF DIBBS Grant ACI-1443054.

REFERENCES

[1] Pankaj K. Agarwal, Alon Efrat, Shashidhara K. Ganjugunte, David Hay, Swaminathan Sankararaman, and Gil Zussman. 2013. The Resilience of WDM Networks to Probabilistic Geographical Failures. *IEEE/ACM Trans. Netw.* 21, 5 (Oct. 2013), 1525–1538. DOI: http://dx.doi.org/10.1109/TNET.2012.2232111

[2] Animashree Anandkumar, Avinatan Hassidim, and Jonathan Kelner. 2011. Topology Discovery of Sparse Random Graphs with Few Participants. *SIGMETRICS Perform. Eval. Rev.* 39, 1 (June 2011), 253–264. DOI: http://dx.doi.org/10.1145/2007116.2007146

[3] Yehiel Berezin, Amir Bashan, Michael M. Danziger, Daqing Li, and Shlomo Havlin. 2015. Localized attacks on spatially embedded networks with dependencies. *Scientific Reports* (2015).

[4] N. G. Duffield, J. Horowitz, F. L. Presti, and D. Towsley. 2002. Multicast topology inference from measured end-to-end loss. *IEEE Trans. Inf. Theory* (2002).

[5] M. Faloutsos, P. Faloutsos, and C. Faloutsos. 1999. On power-law relationships of the internet topology. In *SIGCOMM*, Vol. 29. 251–262.

[6] Peter Grnwald. 2005. A Tutorial Introduction to the Minimum Description Length Principle. In *Advances in Minimum Description Length: Theory and Applications*. MIT Press.

[7] Peter Grünwald. 2004. A tutorial introduction to the minimum description length principle. In *Advances in Minimum Description Length: Theory and Applications*. MIT Press.

[8] Herbert W. Hethcote. 2000. The Mathematics of Infectious Diseases. *SIAM Rev.* 42, 4 (2000), 599–653. DOI: http://dx.doi.org/10.1137/S0036144500371907 http://www.math.rutgers.edu/ leenheer/hethcote.pdf.

[9] X. Jin, W. Yiu, S. Chan, and Y. Wang. 2006. Network Topology Inference Based on End-to-End Measurements. *IEEE Journal on Selected areas in Communications* (2006).

[10] M. Marathe and A. Vullikanti. 2013. Computational Epidemiology. *Commun. ACM* 56, 7 (2013), 88–96.

[11] J. Ni, H. Xie, S. Tatikonda, and Y. Yang. 2010. Efficient and Dynamic Routing Topology Inference From End-to-End Measurements. *IEEE/ACM Transactions on Networking* (2010).

[12] B. Aditya Prakash, Jilles Vreeken, and Christos Faloutsos. 2012. Spotting Culprits in Epidemics: How many and Which ones?. In *ICDM*.

[13] B. Aditya Prakash, Jilles Vreeken, and Christos Faloutsos. 2013. Efficiently Spotting the Starting Points of an Epidemic in a Large Graph. *Knowledge and Information Systems Journal* (2013).

[14] J. Rissanen. 1985. Minimum Description Length Principle. In *Encyclopedia of Statistical Sciences*, S. Kotz and N. L. Johnson (Eds.). Vol. V. John Wiley and Sons, New York, 523–527.

[15] Hiroshi Saito. 2015. Analysis of Geometric Disaster Evaluation Model for Physical Networks. *IEEE/ACM Transactions on Networking* (2015).

[16] David J Schwab, Robijn F Bruinsma, Jack L Feldman, and Alex J Levine. 2010. Rhythmogenic neuronal networks, emergent leaders, and k-cores. *Physical Review E* 82, 5 (2010), 051911.

[17] Devavrat Shah and Tauhid Zaman. 2010. Detecting sources of computer viruses in networks: theory and experiment. In *Proceedings of the ACM International Conference on Performance Evaluation (SIGMETRICS)*. 203–214.

[18] Shashidhar Sundareisan, Jilles Vreeken, and B. Aditya Prakash. 2015. Hidden Hazards: Finding Missing Nodes in Large Graph Epidemics. In *Proc. of SIAM International Conference on Data Mining (SDM)*.

[19] Ye Xia and David Tse. 2006. Inference of Link Delay in Communication Networks. *IEEE Journal on Selected areas in Communications* (2006).