Network Tomography - Sparse Graph Reconstruction

Sufficient Conditions

A Project Report

submitted by

ABISHEK. S

in partial fulfilment of the requirements
for the award of the degree of

MASTER OF TECHNOLOGY and BACHELOR OF TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

June 2013

THESIS CERTIFICATE

This is to certify that the thesis titled **Network Tomography - Sparse Graph Reconstruction (Sufficient Conditions)**, submitted by **Abishek S**, to the Indian Institute of Technology, Madras, for the award of the degree of **Dual Degree**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. 1
Research Guide
Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: 13th June 2013

ACKNOWLEDGEMENTS

I take this opportunity to thank everyone who directly or indirectly helped me complete my project.

I thank my advisor Prof. Radhakrishna Ganti for patiently guiding me throughout this project right from the formulation of the problem till the very end. His way of developing a problem and probing it with the right questions was a very effective tool which I hope to imbibe. I benefited hugely from his passion for academic creativity and rigor which helped shaping the project and its outcome. I will deeply cherish the hours of deep discussions on the huge white-board in his office. After every discussion, I would leave his office having gotten a much better clarity and insight. He is also instrumental in guiding me into the next step of my life outside IITM for a PhD. For this again, I thank him sincerely for agreeing to write recommendation letters for Graduate school applications.

I also thank all the other professors in the Electrical Engineering department who instilled in me the curiosity and hunger to delve deeper and understand the big-picture. This amazing experience as a dual degree student in this department has led me to make a career choice to go ahead and pursue a PhD.

Life at insti was made truly memorable due to the wonderful experiences I shared with my friends here. I take this opportunity to thank my friends in the hostel wing *Wingsutra* -(Abhitesh, Aksh, Akshay, Ashish, Bharath, Bhargava, Dilraj, Naseef, Nikhil, Numaan, Sashikanth, Swostik, Siddarth, Suraj, Vinay). They introduced me to a whole new world with their diversity in cultural and regional backgrounds. I am deeply grateful to them in helping me broaden my view and help gain perspective in life through examples of how they handled various situations and their different attitudes in approaching life. Living with them has definitely been one of my greatest learning opportunity and I feel very fortunate to have had their company.

I would also like to thank '*Gangs of VLSIpur*' (Bharath, Bhargava, Numaan, Prasad, Siddarth, Sudarshan, Vignesh) for being an amazing set of peers to hang out with in the department. I would like to think that I achieved a fuller understanding of the subjects through the various sometimes heated discussions we had. I am very grateful to them as they put up with all my non-sense and for providing an extremely jovial and fun environment to be in. I will always cherish the Tea-shop sessions which would almost always be the most creative discussions of the day.

I would also like to thank the members of the *Networks and Stochastic Systems Lab* (Anjan, Arjun, Aseem, Gopal, Sudharsan and Varsha) for providing a very intellectual atmosphere to be in. The discussions and the arguments on almost any topic under the sun in the lab are some of the fun things I shall definitely miss. I very much enjoyed working here and the lab was like a secondary hostel where on occasions I have spent most parts of the week in the lab.

Last but definitely not the least, I would like to thank my family for everything they have done so far in my life. My parents are true examples of hard working sincere people and that attitude has helped me on numerous occasions in my insti life. I thank my younger brother who despite having his own busy schedules was always there to help me out with a lot of things. I thank my grand-parents for accommodating me and pampering me whenever I visit them. Their support as local guardian has been very instrumental in my life at IITM.

ABSTRACT

KEYWORDS: Tomography, Erdös-Renyi Random graphs, Edit Distance

Graph reconstruction is the problem of estimating the graph (the adjacency matrix) using measurements that can be obtained from the graph. This problem is also broadly referred to as topology discovery or network inference in literature. The required output of any topology inference algorithms is a network structure (the adjacency matrix) of the underlying network which is unknown. In doing so, these algorithms will require some data about the network which are usually measurements generated by running some experiments on the underlying network.

In this thesis, the information about the network is captured through end-to-end shortest path delays between a subset of nodes called *participants*. All participants send probe packets to each other participant through the ***shortest path*** and the corresponding packet delays are noted. These shortest path delay information is the input considered to the estimation algorithm.

Also, the analysis and performance in this thesis is with regard to only reconstructing and inferring topology of *sparse graphs*. That is it is known apriori that the network structure to be inferred is sparse in the number of links. The definition of sparse in this context means that the number of edges in the graph is $\mathrm{O}(V)$. The sparse graphs are statistically modeled using the well studied *Erdös-Renyi Random graphs*.

The main results of this thesis is in characterizing the minimum number of participants required for near-accurate reconstruction of the sparse graph. The analysis aims to state a ***minimal sufficient condition*** under which the tomography of sparse graphs can be performed accurately almost always.

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

BFS	Breadth First Search
ICMP	Internet Control Message Protocol

NOTATION

$G(n, p)$	Erdös-Renyi Random graph ensemble with parameters n and p.
n	The number of nodes in the graph
p	Probability of each edge in the Erdös-Renyi Random graph ensemble
$\frac{c}{n}$	Edge probability in the sparse graph ensemble
m	The number of participants.
n^{α}	The number of participants i.e $n^{\alpha} = m$.

CHAPTER 1

INTRODUCTION

Network Tomography or Topology Inference refers to the task of learning the internal characteristics of the system based on external observable measurements. Thus in terms of networks, the main output of topology inference algorithms is the underlying graph or network structure from which observations are made.

The inference of the underlying network is crucial and a challenging task. Topology inference has vast applications ranging from computer and communication networks to social networks. For instance, in computer or communication networks, the knowledge of the underlying topology can be used for routing especially in ad-hoc networks where there is no explicit server with the entire topology knowledge; evaluating the resilience of the network to node/link failures Kandula *et al.* (2005); for network traffic monitoring and prediction Eriksson *et al.* (2007) or to infer the sources of viruses and other malicious content on the network Shah and Zaman (2010). With regards to social networks, network structure is useful in studying and understanding many characteristics of the society such as identification of hierarchy and community structure Fortunato (2009); prediction of information flow Wu *et al.* (2004) and to evaluate the possibility of information leakage in anonymous social networks. A useful commercial application of topology inference in the context of social networks is to choose sites for percolating information flow which can be exploited heavily by advertisers wanting to push information to all members of the networks by minimizing the amount of initial sites chosen for percolation.

In computer networks and on the internet, the most popular tool to perform network inference is based on 'traceroute' and 'mtrace' which generates path information between pairs of nodes on the internet. However, one main disadvantage of these tools is that they require the cooperation of the intermediate nodes and routers to generate message using the 'Internet Control Message Protocol (ICMP). But of late due to privacy and security concerns Yao *et al.* (2003) a lot of the routers block traceroute requests thus making topology inference on general networks difficult.

Thus what is needed is a method of topology inference without explicit cooperation from all the nodes in the network but only cooperation of a few nodes. This method is what is referred to as Network Tomography in the literature. Tomography approaches uses indirect measurements such as delays of packet transmissions to infer the network. In this project, the system model so employed is such that the data to reconstruct is generated by mechanisms involving no explicit cooperation of the intermediate nodes apart from doing their regular duty such as forward packets on the appropriate link based on the receiver address.

To study the subject of tomography, this project focuses only on the reconstruction of sparse graphs instead of any arbitrary network. The modeling and analysis is based on statistical modeling of network, i.e the topology is assumed to come from a known statistical distribution and the measurements are generated by some known statistical process. The system model and the exact problem setup is explained in the following chapter.

The thesis is organized as follows. Chapter 2 deals with the model chosen for the system and the exact problem setup. Chapter 3 established prior results in this area and frames the central question this project tries to address. Chapter 4 is the solution outline of the problem and its analysis. Chapter 5 is the concluding chapter that deals with the inference from this project and some unanswered questions in this project.

CHAPTER 2

SYSTEM MODEL

This chapter explains the system model and the problem setup.

2.1 Erdös-Renyi Random Graph Model

Network Tomography in the context of this project only concerns the reconstruction of sparse random graphs. The statistical model for the sparse graph used in this thesis is the Erdös-Renyi Random Graph ensemble. The assumption in the rest of the thesis is that the unknown network topology is drawn from a known ensemble of the Erdös-Renyi Random Graph.

The Erdös-Renyi Random Graph is the simplest class of random graphs. It is characterized by two parameters n and p. The notation for the random graph ensemble is $\mathbf{G}(n, p)$. In this notation, n denotes the number of nodes in each graph in the ensemble and p denotes the probability for an each edge to be present in a graph. Thus a sample from this ensemble consists of a graph with n nodes and each of the possible $\binom{n}{2}$ edges occur independently with probability p.

A sequence of graphs are called sparse if the average node degree does not scale with the number of nodes, i.e the total number of edges is of order $\mathrm{O}\big(n\big)$ where n is the number of nodes. The random graph distribution sampled from $\mathbf{G}(n, \frac{c}{n})$ is almost always sparse as the average node degree is c a constant independent of n. Thus the random graph ensemble from now on in this thesis refers to the Erdös-Renyi graph distribution with parameters n and $\frac{c}{n}$.

It is a well known fact that the sparse Erdös-Renyi random graphs exhibit a phase transition with respect to the parameter c. When $c > 1$, there is one giant connected component having $\theta(n)$ nodes for almost every graph in the ensemble while all other connected components have a maximum of $\mathrm{O}\big((\log n)\big)$ nodes. On the other hand if

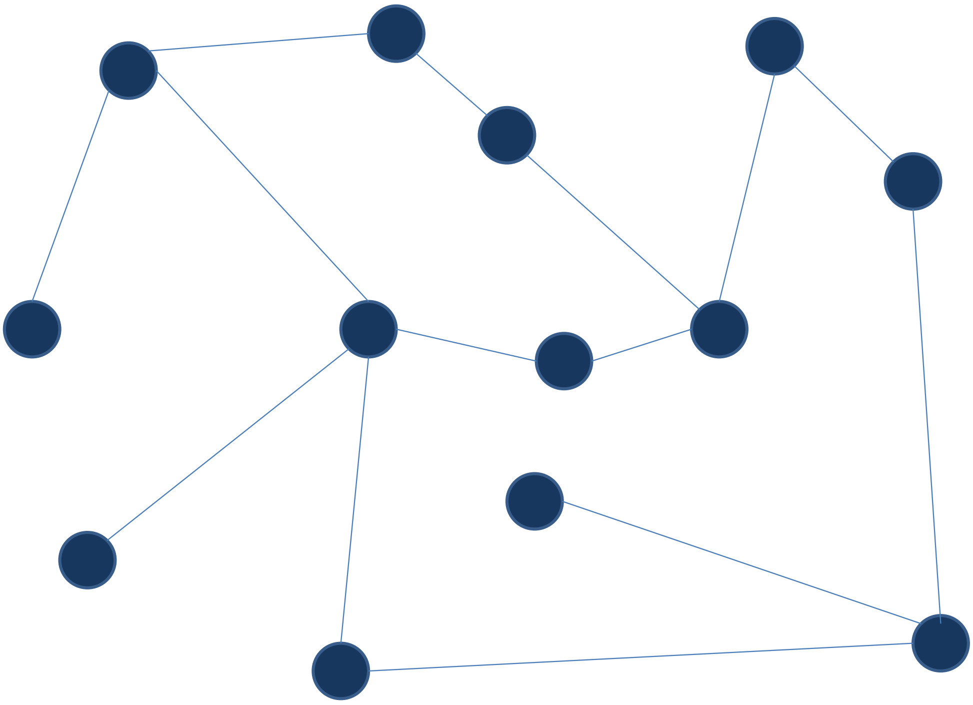

Figure 2.1: An example of a graph

$c < 1$, there will have no giant component i.e the maximum size of the largest connected component will be $\mathrm{O}\big((\log n)\big)$ for almost every graph in this ensemble. Thus $c = 1$ is a threshold for connectivity. The detailed proofs and statements can be found in Bollobás (2001).

Thus in this thesis, it is assumed that $c > 1$ is a constant that is independent of n. Furthermore, topology discovery from now on will refer to the task of identifying this giant component.

2.2 Generating Measurements

The measurement data for the reconstruction is generated by making a certain subset of volunteer node called 'participants' ping each other. The volunteering process can be modeled by assuming that each of the n nodes will with some probability agree to be a 'participant' independently of the other nodes.

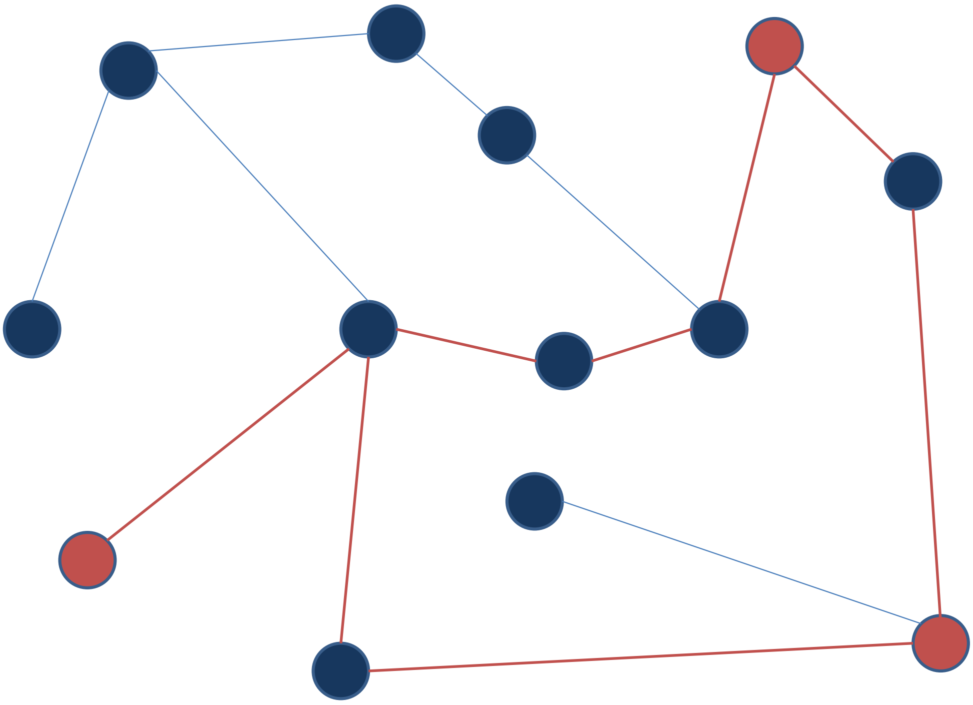

Figure 2.2: An illustration of how the measurements are generated

Once the participants are chosen, each of them ping each other through the shortest path and collect the delays. That is if there are m participants chosen, each of them pings each other i.e a total of $\binom{m}{2}$ pings are done on the network. Each of the ping travels through the shortest path between the respective sender and receiver. At the end, all the $\binom{m}{2}$ shortest path delays are given as input to the topology discovery algorithms.

The fact that the shortest path delays are additive is used in the reconstruction. Ideally after reconstruction one is interested in the individual edge path delay from the aggregated additive delays. Also in this model, it is assumed that the nodes themselves add no delays and all the delays are due to the edges.

For instance in the figure 2.2, there are three participants and hence a total of 3 measurements. Each measurement is the sum of delays on the shortest path links between the participants.

Typically the participants are treated as resources and the interest is in using as few participants as possible. Thus the main requirement is that the number of participants be very small compared to the total number of nodes i.e $m = n^{\alpha}$ where $\alpha < 1$.

2.3 Performance Metric

The performance metric used to quantify the 'goodness' of the algorithm is a metric called 'Edit Distance'. The Edit distance between two graphs is defined as follows.

Definition 1. *Edit Distance: Let F and G be two graphs with $\mathbf{A_F}$, $\mathbf{A_G}$ as the adjacency matrices. Let V be the set of labeled vertices in both graphs. Then the edit distance between F and G is*

$$\Delta(F, G; V) := \underset{\pi}{min} ||\mathbf{A_F} - \pi(\mathbf{A_G})||_1 \tag{2.1}$$

where π is the permutation of the unlabeled nodes.

The permutation over the unlabeled nodes is present in the definition because a graph and its isomorphisms are equivalent under the reconstruction procedure. This is a valid model to put as there is no node label information for the non-participating nodes in the measurement data.

The objective of the reconstruction algorithm is to output a graph which has *sub-linear* edit distance with the original graph. That is the algorithm is said to be correct if the graph it outputs $\hat{G}$ and the original graph G differ in edit distance by at most n^{ρ} for some $\rho < 1$.

We are interested in designing algorithms that can give sub-linear edit distance guarantees not linear or higher orders of n because of the following observation by Anandkumar *et al.* (2012).

Anandkumar *et al.* (2012) showed that any two random graphs with high probability are separated linearly in edit-distance i.e the average edit distance between any two random graphs sampled from $\mathbf{G}(n, \frac{c}{n})$ is $(0.5c - 1)n$ which is $\theta(n)$. Thus if the objective was to reconstruct only upto linear edit distance, then any graph randomly sampled from the distribution $\mathbf{G}(n, \frac{c}{n})$ will suffice. Thus, the interesting regime is to design algorithms that can give sub-linear edit distance guarantees.

2.4 Model Summary

The summary of the problem setup is as follows.

- The sparse graph to be estimated comes from the distribution $\mathbf{G}(n, \frac{c}{n})$. c is a constant assumed to be known apriori.
- Given the graph the participant nodes are chosen at random and independent of other nodes.
- Once the participants are chosen, each participant pings every other participant through the *shortest paths*. These shortest path delays are then fed as the input to the topology inference algorithm.
- Using this input, the aim is to output a graph that differs from the original graph atmost upto sub-linear edit distance.

CHAPTER 3

CENTRAL QUESTION ADDRESSED

This chapter outlines the main question that is targeted through this project. The following sub-section establishes the prior results and questions answered in this field.

3.1 Prior Work

The most significant contribution to this field was from Anandkumar *et al.* (2012). They proved the following results -

- The number of participants required under this model to reconstruct accurately upto sub-linear edit distance is $\mathrm{O}\big(n^{\frac{5}{6}}\big)$
- Probability of reconstruction error will tend to one for **any** choice of participants if the number of participants $m < \mathrm{O}\big(\sqrt{n}\big)$.

The first result was shown by proposing a reconstruction algorithm and showing that the algorithm achieves sub-linear edit distance with high probability if the number of participants is greater than $\mathrm{O}\big(n^{\frac{5}{6}}\big)$.

The second result is a result on the necessary number of participants. The result makes it clear that choosing any fewer than $\mathrm{O}\big((\sqrt{n})\big)$ participants can never give sub-linear edit distance.

3.2 Relaxations

In this project, further relaxations were made as compared to the model in Anandkumar *et al.* (2012). Specifically, the following relaxations are made -

- The number of nodes n is assumed to be known.

- All edge weights are equal and known.

The first assumption on the total number of nodes turns out to be a not so severe relaxation. This number can usually be estimated fairly accurately and quickly using other well studied techniques.

The second assumption stating that the edge weights are all equal and known is a good model for the case when only hop counts are known instead of the actual delays. For instance in computer networks, the TTL field of the packets can give an estimate of the number of hops between the transmitter and receiver. Therefore in the situation when only the number of hops between two nodes are available, the model can be thought of as a random graph with all edge weights equal to one. This model is what was studied in this project.

3.3 Main Question

With the prior work as the background and the relaxations made, the central question this project explored was the following.

What is the minimal sufficient number of participants needed to achieve sub-linear edit distance with high probability ?

A sufficient number of participants as proposed by Anandkumar *et al.* (2012) is $\mathrm{O}\big((n^{\frac{5}{6}})\big)$. The question this project explored was that whether $\mathrm{O}\big((n^{\frac{5}{6}})\big)$ is the best possible in terms of the number of participants or can one do as well with fewer participants?

The necessary number of participants as proved by Anandkumar *et al.* (2012) is $\mathrm{O}\big((\sqrt{n})\big)$. Can this also be a sufficient number of participants to achieve reconstruction upto sub-linear edit distance ?

To answer this question, an exponential algorithm was proposed and its requirements in terms of the number of participants was analyzed. If we are able to show that

$\mathrm{O}\big((\sqrt{n})\big)$ number of participants is sufficient, then it is also the minimal sufficient as $\mathrm{O}\big((\sqrt{n})\big)$ participants is also a necessary number.

In the rest of the thesis, an algorithm and its analysis is outlined and it can be concluded that ***maybe*** $\mathrm{O}\big((\sqrt{n})\big)$ number of participants can suffice for reconstruction upto sub-linear edit distances.

CHAPTER 4

SOLUTION and ANALYSIS

This chapter proposes an exponential algorithm and its performance analysis.

4.1 Algorithm

The main data input to the algorithm is the $\binom{m}{2}$ vector of the shortest distance hop counts. In the description of the algorithm, denote by the set $\mathbf{G}$ as the set of all graphs on n nodes. The graphs in $\mathbf{G}$ are arranged in the increasing number of edges i.e for two graphs on n nodes A and B, if the number of edges in A is less than the number of edges in graph B, then the graph A occurs before B in the ordered set $\mathbf{G}$. Graphs having the same number of edges are however ordered relatively at random.

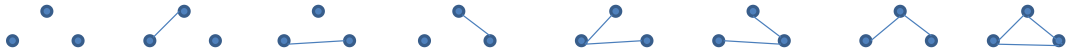

Figure 4.1: An example of the ordered set $\mathbf{G}$ set when $n = 3$. The Graphs with fewer edges have a lower index value as compared to graphs with higher number of edges

The actual algorithm is as follows -

Algorithm 1 Graph Estimator

```
procedure ESTIMATOR($m, y$)
    for each graph $G_i$ in $\mathbf{G}$ do
        Take measurements $y^{'}$ using the $m$ participants on $G_i$.
        if $y^{'} == y$ then
            Output $G_i$ as the estimated graph.
            Break
        end if
    end for
end procedure
```

4.2 Analysis

This aim of this chapter is to identify circumstances under which the proposed algorithm would make an error and compute the probabilities of those events.

In this section the notation G_n refers to the original graph whose topology is to be inferred.

The algorithm will make an error if there exists another graph $G^{'} \neq G_n$ in which the number of edges is less than or equal to then umber of edges in G_n and the experiment with the chosen participants give the same measurement in $G^{'}$ as in G_n.

The error events can be broken into the two following classes.

1. There exists a graph $G^{'}$ with the number of edges in $G^{'}$ being strictly lesser than that in G_n and $G^{'}$ satisfies the measurements.
2. There exists a graph $G^{'}$ having the same number of edges as G and satisfying the measurements and $G^{'}$ occurs before G_n in the set $\mathbf{G}$.

From the nature of the algorithm, it is clear that it will not make a mistake if all other graphs $G^{'}$ that satisfy the measurements have greater number of edges than G_n (the original graph).

For analyzing the first type of error event, one can ask the following question

When can the number of edges in G_n be reduced without affecting the measurements ?

It turns out the number of edges can be reduced at a non-participating node whenever it is 'bad'.

A non-participating node is defined as 'bad' if it has two edges incident on it that are part of the same set of shortest path pings.

Every shortest path ping travels through a some subset of edges on the graph. If there exists two edges incident on a non-participating node such that both of the edges are part of the same subset of the $\binom{m}{2}$ shortest path pings, then that non-participating node is defined as 'bad'.

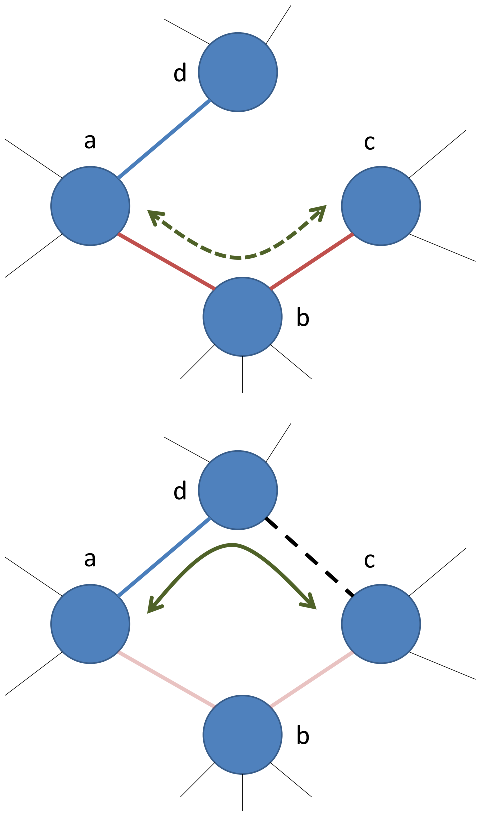

Figure 4.2: An illustration of the bad node

In 4.2, node b is the 'bad node'. The edges on the top figure $a - b$ and $b - c$ marked in red have the same set of shortest path pings traveling through them. Thus the total number of edges in the graph can be reduced as shown in the figure on the bottom by deleting edges $a - b$ and $b - c$ and adding a new edge $d - c$.

Thus the number of participants chosen must be atleast high enough so that probability that a node is 'bad' can be made smaller than a required tolerance level.

Probability of Algorithm Error

This sub-section outlines how to extend the probability that a node is 'bad' to the probability of the algorithm making an error.

One observation to make is that every 'bad' node can contribute at-most $\frac{3}{2}d$ where d is the degree of the bad node. Thus the total edit-distance can be bounded by the knowledge of the number of bad-nodes and their degrees. The algorithm is said to make an error if the edit-distance exceeds sub-linear limit in the number of nodes.

The expected edit-distance can be calculated by using the fact that the node-degree of sparse random $\mathbf{G}(n, \frac{c}{n})$ is Poisson distributed with parameter c and applying the union bound over all non-participants.

However, it turns out that to compute that the probability that a node is 'bad' is quite difficult. But an upper bound to it can be computed by making the following observation.

Let $\mathbf{E_1}$ be the event that a random node is 'bad' , i.e there exists a pair of edges incident on it carrying the *same* set of flows.

Define another event $\mathbf{E_2}$ which denotes the event that any randomly chosen node along with any two edges incident on it does ***not*** carry a common shortest path ping.

Hence, it can be seen that

$$E_1 \subset E_2$$

and therefore

$$Pr(E_1) \leq Pr(E_2)$$

Thus an upper bound on $Pr(E_2)$ will also be an upper bound on $Pr(\text{node is 'bad'})$.

And it also turns out that computing an upper bound on $Pr(E_2)$ is simple.

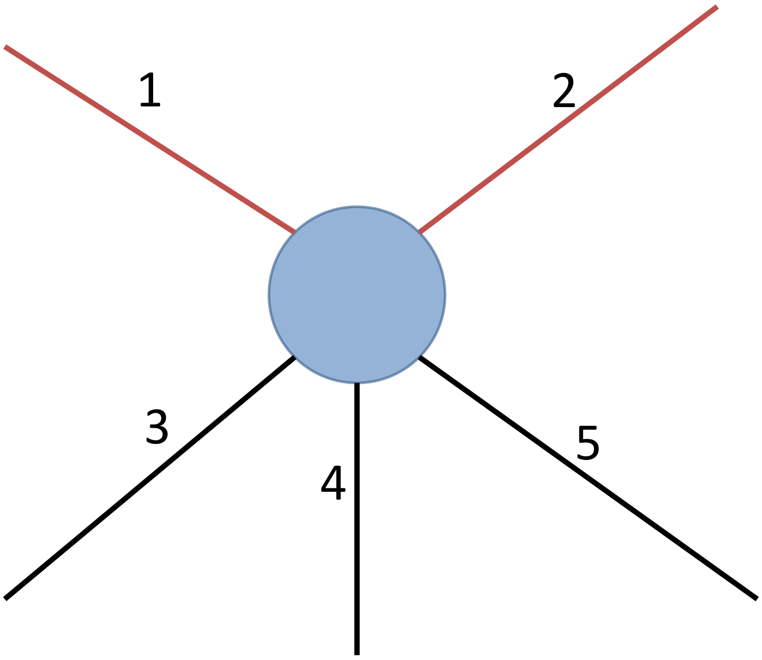

Figure 4.3: An illustration of why $E_1 \subset E_2$

The figure 4.3 illustrates why whenever E_1 occurs, E_2 definitely occurs. In 4.3, edges 1 and 2 carry the same set of shortest path pings and hence make the node bad. Thus this also means that both edge 1 and 2 share no common shortest path pings with the edges 3, 4 and 5.

To compute an upper bound on $Pr(E_2)$, the following lemma is useful.

Lemma 1. *In a random graph* $\mathbf{G}(n, \frac{c}{n})$*, the inter-vertex distance distribution follows* $Pr(d_{ij} > l) = \exp\left(-\frac{c^l}{n}\right)$ *where* d_{ij} *refers to the shortest distance between two randomly chosen nodes in the graph.*

Proof. This lemma is proved in Newman (2010) and is reproduced here .

Consider two vertices i and j. Draw two 'balls' or neighborhood around i and j consisting of vertices with distances up to and including s and t respectively. Consider the set of vertices on the 'surface'(i.e. the most distant vertices) of either of the balls. If there exists no edge between the two surfaces, then the distance between i and j is necessarily greater than or equal to $s+t+1$. Thus a necessary and sufficient condition for $d_{ij} > s+t+1$ is that there exist no edge between the two 'surfaces'. Thus $Pr(d_{ij} >$

$s + t + 1)$ is the probability that there exists no edge between the two surfaces.

The average number of nodes on the surface of the ball from i at distance s is c^s and the number of nodes at the surface of the ball from j at distance t is c^t. Thus there are $c^s \times c^t$ pair of vertices and each pair can be connected with probability c/n.

Thus the probability that there exists no edge is $(1 - \frac{c}{n})^{c^{s+t}}$.

Replacing $s + t$ by l, the probability expressions becomes

$$Pr(d_{ij} > l) = \exp{(-\frac{c^l}{n})} \tag{4.1}$$

□

One interpretation of this result is that the distance between any two randomly chosen nodes is $\mathrm{O}\big(\log n / \log c\big)$ with high probability.

Using the above result, an upper bound on the probability of event E_2 can be computed.

To compute an upper bound on $Pr(E_2)$, a lower bound on the complimentary event $\bar{E_2}$ is computed.

The event $\bar{E_2}$ denotes that every pair of edge incident on a non-participant has a common shortest path ping through them.

To compute for a single node, a *breadth first tree* is drawn about the node.

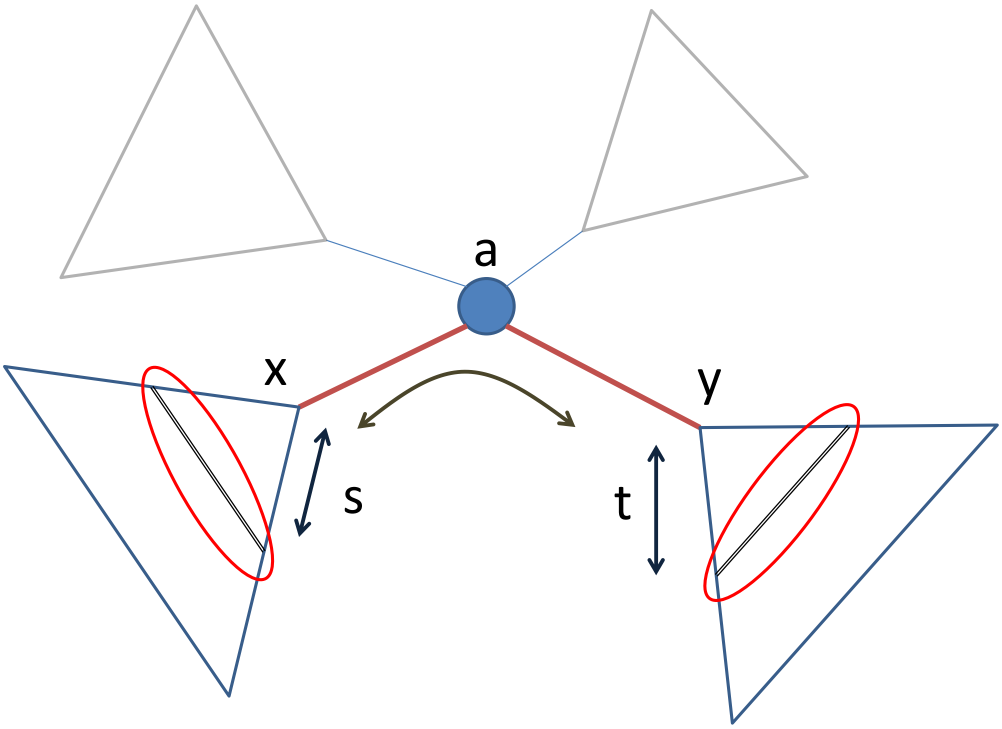

Figure 4.4: The breadth first tree rooted at node a

To compute an upper bound, the probability of one event that necessarily produces a common shortest path ping is evaluated.

For instance in figure 4.4, the edges of interest is $a - x$ and $a - y$ and want to determine at-least one scenario in which the edges $a - x$ and $a - y$ share a common shortest path ping.

To compute a lower bound on $Pr(\bar{E}_2)$, the probability of a particular event that will imply that atleast one flow exists through $a - x$ and $a - y$ is computed. The following event ensures that a flow will always exist through $a - x$ and $a - y$.

If there exists two participants at depth of s in the sub-tree(of the BFS tree rooted at a) from node x and at depth t in the sub-tree of the BFS tree rooted at a) from node y and they are connected by a shortest path whose length is greater than or equal to $s + t + 2$, then $a - x$, $a - y$ has a shortest path flow. The probability of $\bar{E}_2$ can then be estimated by summing the probability of the above event over all possible values of s and t. Thus, a lower bound on $\bar{E}_2$ can be found by computing the probability of the above event for one particular value of s and t.

For the calculation, the value of $s = t = \frac{\log n}{2 \log c(1+\epsilon)}$ is used. Denote by the event B that at-least one shortest path flow exits through $a - x$ and $a - y$ originating from a participant which are located at depth s of the sub-tree from node x and at depth t of sub-tree from node y. Clearly since B does not sum over all possible values of s and t,

$$Pr(\bar{E_2}) \geq Pr(B)$$

Thus $Pr(B)$ serves as a lower bound on $Pr(\bar{E_2})$.

The BFS tree for small depths can be modeled as a Branching process with offspring distributed as Poisson with parameter (c-1). This branching process is also referred to as the Galton-Watson process.

At the depth where $s = t = \frac{\log n}{2 \log c(1+\epsilon)}$, the number of nodes of the Galton Watson process can be approximated as $Z_s = (c-1)^s$ and $Z_t = (c-1)^t$ as s and t are large (increasing function of n). (Here Z_i denotes the number of nodes at depth i in the breadth first tree). Thus the number of nodes at this depth in the random graph is

$$\begin{aligned} Z_s = Z_t &= (c-1)^{\frac{\log n}{2 \log c(1+\epsilon)}} \\ &= n^{\frac{1}{2(1+\epsilon) \log_{c-1} c}} \end{aligned}$$

Since Z_s is much smaller than the n (the total number of nodes in the graph), the locally tree like property of the graph holds true implying that the Galton-Watson model is an accurate model of the BFS tree at this depth.

The probability that there will exists at-least one participant at depth $\frac{\log n}{2 \log c(1+\epsilon)}$ in the breadth first tree is given by

$$1 - \left(1 - \frac{n^{\frac{1}{(2(1+\epsilon)})}}{n}\right)^{n^{\alpha}}$$

where n^α is the number of participants.

The probability that the shortest distance between two nodes is greater than l is given by from lemma 1

$$Pr(d > l) = \exp\left(\frac{-c^l}{n}\right)$$

In this calculation, $l = \frac{\log n}{\log c(1+\epsilon)}$.

Since the participants are chosen independently of the underlying graph, the probability of event B is given by

$$Pr(B) = \left(1 - \left(1 - \frac{n^{\frac{1}{(2(1+\epsilon)})}}{n}\right)^{n^\alpha}\right)^2 \exp\left(\frac{-c^{\frac{\log n}{\log c(1+\epsilon)}}}{n}\right) \tag{4.2}$$

The value of α that makes $\lim_{n\to\infty} Pr(B) \to 1$ is $\alpha > 1 - \frac{1}{2\log_{c-1} c}$.

$$Pr(\bar{E_2}) \geq Pr(B),$$
$$1 - Pr(\bar{E_2}) \leq 1 - Pr(B)$$

Thus,

$$Pr(E_2) \leq 1 - Pr(B) \tag{4.3}$$

Therefore, $\forall \alpha > 1 - \frac{1}{2\log_{c-1} c}$, $Pr(E_2)$ tends to zero.

To compute an upper bound on the probability that a node is 'bad', the event B must be summed over all edge pairs incident on the node using union bound, i.e,

$$Pr(\text{node is bad}) \leq \sum_{d=3}^{\infty} \binom{d}{2} \left(Pr(\bar{E_2})\right) c^d e^{-c}/d! \tag{4.4}$$

which also tends to zero for all $\alpha > 1 - \frac{1}{2\log_{c-1} c}$.

Moreover, $Pr(\text{node is 'bad'})$ decays to zero as $\exp(-n^{\gamma})$ where

$\gamma = \alpha - \left(1 - \frac{1}{2\log_{c-1} c}\right)$.

This implies that the expected number of bad nodes given by

$n.Pr(\text{node is 'bad'}) = k^{'} n \exp(-n^{\gamma})$ also tends to zero as n tends to infinity.
where $k^{'}$ is a constant depending on c.

The conclusion from this analysis is that the edit distance can be made to decay to zero exponentially fast whenever

$$\boxed{\alpha > \left(1 - \frac{1}{2\log_{c-1} c}\right)}$$

4.3 Possible Gaps in the Analysis

The above analysis takes care that there exists no other graph with edges smaller than the number of edges present in the original graph can generate the same set of measurements. However, the other error event i.e the non-existence of another graph having the same number of edges is not covered in this analysis.

The other question to ask is does there exists graphs that are non-isomorphic to the original graph satisfying the measurements **and** every pair of edge having a common end point satisfying property B as defined in the previous sub-section?

This question is however unanswered at the moment and is left for future work.

CHAPTER 5

CONCLUSION

This project was an attempt at answering the question as to what is the minimal sufficient number of participants required for sparse graph reconstruction. The approach taken was to outline an exponential algorithm and then analyze the same for determining how high should the number of participants be for reconstructing within sub-linear edit distance with high probability. The conclusion from the analysis was that if the number of participants m is of the form where $m = n^{\alpha}$, then ***maybe*** $\alpha > 0.5$ can suffice. There is still ambiguity as there is no proof as of yet as to all possible error events have been covered in the analysis presented.

5.1 Future Work

One immediate future work is to conclusively establish all the error events of the algorithm proposed in this project.

The other direction of future work is to extend this analysis for weighted graphs. How does on model a weighted graph ? Do we sample a random structure and then assign random weights or does a joint distribution of the structure and the weights a more accurate representation of the real world scenario ? This modeling can also help in ordering the graphs in the set $\mathbf{G}$.

The other important direction of future work is in establishing polynomial run-time algorithms requiring only $\alpha > 0.5$ as the order of the number of participants. This project although outlined an algorithm, it is doubly exponential and can only be used as an argument to prove existence of algorithms and nothing more. Thus can one do better in polynomial time requiring fewer than $\mathrm{O}\big((n^{\frac{5}{6}})\big)$ participants?

REFERENCES

1. **Anandkumar, A.**, **A. Hassidim**, and **J. Kelner** (2012). Topology discovery of sparse random graphs with few participants. *Random Structures & Algorithms*.

2. **Bollobás, B.**, *Random graphs*, volume 73. Cambridge university press, 2001.

3. **Eriksson, B.**, **P. Barford**, **R. Nowak**, and **M. Crovella**, Learning network structure from passive measurements. *In Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07. ACM, New York, NY, USA, 2007. ISBN 978-1-59593-908-1. URL `http://doi.acm.org/10.1145/1298306.1298335`.

4. **Fortunato, S.** (2009). Community detection in graphs. *CoRR*, **abs/0906.0612**.

5. **Kandula, S.**, **D. Katabi**, and **J.-P. Vasseur**, Shrink: A Tool for Failure Diagnosis in IP Networks. *In ACM SIGCOMM Workshop on mining network data (MineNet-05)*. Philadelphia, PA, 2005.

6. **Newman, M.**, *Networks: An Introduction*. Oxford University Press, Inc., New York, NY, USA, 2010. ISBN 0199206651, 9780199206650.

7. **Shah, D.** and **T. Zaman** (2010). Detecting sources of computer viruses in networks: theory and experiment. *SIGMETRICS Perform. Eval. Rev.*, **38**(1), 203–214. ISSN 0163-5999. URL `http://doi.acm.org/10.1145/1811099.1811063`.

8. **Wu, F.**, **B. A. Huberman**, **L. A. Adamic**, and **J. R. Tyler** (2004). Information flow in social groups. *Physica A: Statistical and Theoretical Physics*, **337**(1-2), 327–335. URL `http://dx.doi.org/10.1016/j.physa.2004.01.030`.

9. **Yao, B.**, **R. Viswanathan**, **F. Chang**, and **D. Waddington**, Topology inference in the presence of anonymous routers. *In In IEEE INFOCOM*. 2003.