A Scalable Probing-based Approach for Congestion Detection using Message Passing

Rajesh Narasimha, Souvik Dihidar, Chuanyi Ji and Steven W. McLaughlin
Georgia Institute of Technology, Atlanta, GA 30318 USA
Email: {rajesh, dihidar, jic, swm}@ece.gatech.edu

Abstract-In this paper we consider the theoretical and practical management resources needed to reliably localize congested nodes/links and propose a solution based on decoding linear error control codes. We define the scalability of measurement-based network monitoring and relate this to the problem decoding of linear error control codes on a binary symmetric channel. Our main goal is to minimize the number of probe packets required to fully identify congested nodes and our approach requires a number of measurements that grows linearly with respect to the network size, i.e. the approach is scalable. We provide fundamental limits on the relationship between the number of probe packets, size of the network and the ability to perfectly identify all congested nodes. To identify the congested nodes we construct a bipartite graph, and congestion is inferred using the message-passing algorithm. Simulation results demonstrate the ability to perfectly locate congested nodes using this approach.

1. INTRODUCTION

As the growth rate of the Internet and its traffic continue, the issue of congestion detection and mitigation become increasingly important. The ideal goal is to be able to reliably detect all congested node/links in a large network with little overhead inserted into the network traffic. It is also highly desirable that any congestion detection approach be scalable, in the sense that, as the network become large, the congestion detection method should not become prohibitively costly or complex.

This work investigates detection and localization of persistent congestion in the interior of a network using end-end probe measurements. Persistent congestion is a symptom that occurs due to overload [1], traffic clustering and malicious attacks such as distributed denial of service (DDos) [2] and worm propagation. Long-term or persistent congestion that arises due to bandwidth stealing/flooding of malicious traffic causes the user flows to experience longer delays, higher loss rates and lower throughput and lasts for a longer period of time. Human intervention is necessary to overcome such type of congestion. Our goal is two fold: (a) studying the scalability of the management resources as the growth rate of the number of measurements with the size of a network, and (b) developing an efficient algorithm, which performs the localization using a minimal number of probe packets and computation. In the current approach, accurate localization is desirable without adding overhead to the existing network and which can be easily integrated into the existing network infrastructure. Moreover, end-to-end probing is a necessary ingredient in our scenario as opposed to passive methods.

Researchers have investigated both passive and active (i.e., probing) methods to detect and mitigate congestion. For example, passive measurements can be obtained using network tools that are built into the routers such as SNMP (Simple Network Management Protocol), Netflow and RMON to monitor the internal status of the network. Typically routers have to be polled to collect network statistics such as delay, loss and available bandwidth. To obtain useful monitoring information, large amounts of data need to collected which can be prohibitive if the network size is large. For instance, studies have shown that periodic polling of Cisco-4000 series Netflow-enabled routers on a local network decreased the throughput by as much as 15-20% [3]. The use of end-end probe-measurements avoids polling from internal nodes. In an inter-domain setting, since the domains are managed by multiple ISPs and the service providers do not disclose confidential information about their domains, probing may be the only convenient choice when the interior of a network is not assessable directly [2]. The authors in argue [4] that 80% of the congestion occurs within the access ISP network which are operated at higher utilization levels than other networks. The detection and localization may also be made easier using the end-end measurements as they contain global information of a network. Conventional methods of detecting congestion are by using packet pair correlation [5] where the end-to-end delay/loss are observed at the end hosts. Passive approaches have been investigated in [6], where the authors argue that the passive approach does not generate probe traffic.

Existing tools identify the congested segments, but are not effective in identifying the actual congested nodes/links. The work in [7] proposes a practical method to infer congested segments in real-time based on indirect inference methodology using multiple end-to-end measurements. In [8] , a tool called *Pathneck* is presented that allows end users to efficiently and accurately locate the bottleneck links on a internet path using distributed framework. Unlike other approaches, the work in [3] assumes a network operations center that gathers information on bandwidth and latency using SNMP, RMON/Netflow and explicitly routed IP probe packets, and proposes approximation algorithms to optimize the overhead of the measurement model. A methodology for measurement and classification of bottleneck links based on the investigation of links within a managed domain of a carrier ISP or between neighboring carriers is addressed in [9].

As many inference approaches have been developed, there have only been a few studies quantifying the amount of management resourced needed. A probing scheme which uses *O(nlogn)* measurements based on weighted-set cover algorithms is presented in [10], where *n* is the number of edge routers that form an overlay network. Habib *et al* [2] investigate a monitoring scheme which requires *O(n)* probes, if the congestion is less than 20%. The results show that the algorithm requires $O(n^2)$ probes when the congestion is more than 20%. The scalability is defined and investigated in the context of density estimation for multicast inference of link loss probabilities in [11]

The main contributions of our work are:

- We provide a general definition for scalability extended from [11] : we define it as the growth rate of the number of measurements with respect to the network size for accurate inference given the performance measure and evaluate the scalability of measurement-based network monitoring. In the present framework network monitoring/inference refers to localization of congested nodes/links in the network.

- We devise a methodology to perfectly detect which nodes/links are congested in real-time using probe packets sent from multiple sources to destinations. This is accomplished using scalable edge-based inference techniques where the monitoring functionality is strictly limited to the end-hosts. This kind of indirect inference method based on end-to-end measurements is known as network tomography (see [12] and the references therein). The monitoring approach requires just one bit of payload in the probe packet for congestion localization. Hence, this approach is designed to provide better quality of service to the users without overburdening the network.
- We provide theoretical bounds on the growth rate of the number of measurements by relating the congestion localization in networks to the problem of decoding linear error correcting codes (ECC) over a binary symmetric channel (BSC) in coding theory.

In particular, the congestion localization problem is dealt with in a two-fold fashion. The probing paths and the nodes in the corresponding path form a directed graph [13] where the nodes in the probed path (n') are the parents that influence the end-to-end output. Initially, the number of probing paths, denoted as *m,* is fixed, and we evaluate the maximum fraction (ρ) of the congested nodes that can be detected with zero detection error both in the presence of noiseless and noisy probe outcomes. The output of each probe path corresponds to an observation, and the number of probe paths is the number of measurements. Secondly, we examine the growth rate of the number of measurements as the percentage of congested nodes increases. Scalability is quantified in terms of the ratio of the number of edge routers that are used in the probing experiment to the nodes/links that are present in the probing paths. Furthermore, if we can tolerate a detection error δ , a higher prior (fraction of congested nodes) can be detected. We show that the number of measurements $m \le cn'$, where c is a constant. Although a significant number of algorithms have been proposed on inference techniques, the computational complexity is either unknown or unsuitable for practical implementation. The inference of the nodal states in the present scenario is performed using message-passing algorithm, which is known to have a computational complexity proportional to $O(n \log n)$, n being the block length of the LDPC code. In the case of congestion localization, the computational complexity proportional to $O(n' \log n')$. The scalability results are verified using simulations for varying network sizes.

Graphical models have been widely studied in image restoration and coding problems [14] and is beginning to find applications in networking problems. A closely related approach is the application of factor graphs to estimate the link loss in sensor networks where the construction of the graph is based on link costs and path flows, which are assumed to be independent [15]. Bayesian belief networks have been applied to fault localization and detection problem in [16]. The bipartite graph formulation of the congestion localization problem facilitates the use of message passing algorithm.

The remainder of this paper is organized as follows. The problem description along with the assumptions are presented in section 2. Mapping of the congestion localization to error control coding over BSC are provided in section 3. Scalability and simulation results are discussed in section 4. Conclusions and future work are given in section 6.

2. PROBLEM DESCRIPTION

Given a network, the problem here is to determine the maximum fraction of congested nodes/links that can be detected with the given performance measure. The performance measure is the detection error, and can be either zero or a small value. The congestion localization is accomplished using a chosen *subset* of paths such that monitoring on these paths is sufficient to infer every congested nodes/links in the network based on end-to-end measurements. In terms of the network resources, monitoring on these fixed subset of paths are performed using end-end probe packets. Since the network resources are expensive, it is desirable that the inference of node/link states can be obtained using a minimum set of probes and as the size of the network grows, the growth rate of the number of probes should be linear with respect to the network size.

Definition 1: *The scalability is the growth rate of the number of measurements (probes) that are required with respect to the size of the network to infer the status of the nodes/links with a given detection error* δ .

Consider a graph *G(V,E),* where *V* is the set of nodes and *E* is the set of edges. Assuming all nodes in the network are vulnerable to congestion, each network node has two states, congested or not. Let us define the status of each node as X_i , i being the node index. Suppose there are *n*-nodes in the network then $\boldsymbol{X} = \{X_1, X_2, ...X_n\}$. The status of each node can be defined as

$$X_i = \begin{cases} 1 , & \text{if not congested}, \\ 0 , & \text{if congested}. \end{cases} \quad (1)$$

Hence the vector $\boldsymbol{X} \in \{0,1\}^n$. Probe packets are assumed to be marked with the highest priority, and sent from different sources to destinations, where the destinations correspond to the edge-routers/end-hosts that are equipped with monitoring capability. The probe packet is a normal packet but has a unit bit-field that is initialized to zero and is flipped on its way through the selected probing paths if the node is not congested. When the packet arrives at a congested node, due to its high priority it is just pushed to the beginning of the queue and is passed on to the next node without flipping. Once the packet arrives at the destination its binary bit field is observed.

Consider Fig. 1 where the shaded router is congested. We construct a bipartite graph as shown in Fig. 1, where on one side we have the set of paths the probe takes and on the other side the nodes in those paths. The nodes in a particular path influence the output of that path and hence edges are directed from the nodes in the path to the output. The observation vector is the output of the probe packet bit field collected at the end-hosts. We point out that if *n* is the total number of nodes inside the given IP topology, the number of observations *m* needed is always less than or equal to n' (independent observations), where n' is the number of nodes on the bipartite graph. For $m \le n'$, the proposed approach can detect up to a certain fraction of congested nodes and the maximum fraction being when $m = O(n')$, ensuring zero detection error. We make the following assumptions

1. Underlying IP topology is known.
2. Core routers have the capability to flip the bit in case they are not congested with probability1 (noise is assumed to be zero).
3. Each node gets congested independently of the other nodes.

4. All the edge routers have probing capabilities.
5. The fraction of congested nodes (ρ) is known apriori. Since in reality this is unknown, the message-passing algorithm can infer the status without this information but with a higher detection error.

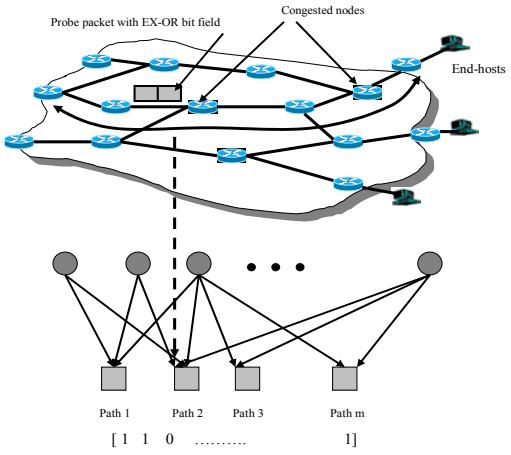

Fig. 1. Representation of congestion on the Internet

Consider the following example. Where the packet path is defined by nodes $\{1,5,78,45,90\}$. For example, let us assume in this path node 5 and nodes 78 are congested, we initialize the bit field in the probe packet to zero. Since nodes 1, 45 and 90 are not congested the bit field is flipped to one at node 1, and then at node 5 the bit remains unchanged and at node 45 the bit is flipped back to zero. Since node 78 is congested, it passes the probe packet and then at node 90 the bit is again flipped to one. So the output of the bit field is one, this is one of the observations in the observation vector y. Since, the probe packet goes through node indices $\{1,5,78,45,90\}$, the output at the edge-router is $X_1 + X_5 + X_{78} + X_{45} + X_{90}$, where ' + ' denotes the binary element-wise EX-OR operation. Here each X_i represents the status of each node as in (1) and hence the output is $X_1 + X_5 + +X_{78} + X_{45} + X_{90} = 1$. Hence, the problem of state observation has been converted to solving a set of linear equations given a set of observations and is of the form

$$A\boldsymbol{x} = \boldsymbol{y} \qquad (2)$$

where A is the routing matrix of size $n \times m$ and has a '1' in the ij^{th} position if node i lies on path j. $\boldsymbol{y}$ is the set of observations measured at the edge routers, which is a column vector of size $m \times 1$ and $\boldsymbol{x}$ denotes the realizations of the random variable $\boldsymbol{X}$ and is of size $n \times 1$.

Usually the dimension of A is large and in general it is not full-rank. Therefore, problem of identifiability exists and several iterative algorithms are proposed [17]. Therefore there are many possible solutions to equation (2), and we choose one of the maximum likelihood solutions where, $\boldsymbol{x}$ has the maximum Hamming weight (maximum number of 1's).

3. RELATION TO ERROR CORRECTING CODES (ECC) FRAMEWORK

3.1. Background

In this section we draw similarities between the congestion localization and the problem of decoding ECC over a binary symmetric channel. Let us consider a binary symmetric channel as in Fig. 2 with input symbols $X \in \{0,1\}$ and error probability ρ , which is also termed as crossover probability. The channel model is shown in Fig. 3, where $\boldsymbol{c}$ is the codeword that is sent through the channel that adds noise vector $\boldsymbol{n}$ and the received codeword is $\boldsymbol{r}$. $\boldsymbol{c}$ is a n-bit codeword, which can be any one of the 2^k possible codewords, k being the message length. This is called the *(n, k)* binary code. For a *(n, k)* binary code the rate R is defined as k/n. From Fig. 3, we have $\boldsymbol{r} = \boldsymbol{c}+\boldsymbol{n}$, where '+' denotes the element

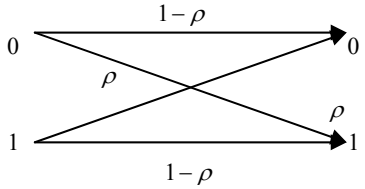

Fig. 2. Binary Symmetric Channel (BSC).

wise binary XOR operation. Here the noise vector $\boldsymbol{n}$ is *iid* with $\Pr\{n_i = 1\} = \rho; \Pr\{n_i = 0\} = 1 - \rho$.

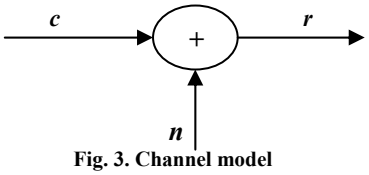

Fig. 3. Channel model

Let us consider a linear error correcting code, which has the following property

$$H\boldsymbol{c} = 0 \qquad (3)$$

where H is the parity-check matrix of size $(n-k) \times n$ and has full row-rank. Given the received codeword $\boldsymbol{r}$ and the condition $H\boldsymbol{c} = 0$, we find the transmitted codeword $\boldsymbol{c}$ and the noise vector $\boldsymbol{n}$. Let $H\boldsymbol{r}=\boldsymbol{d}$ and since $\boldsymbol{r} = \boldsymbol{c}+\boldsymbol{n}$, we have

$$H\boldsymbol{n}=\boldsymbol{d}. \qquad (4)$$

One possible method of obtaining the transmitted codeword $\boldsymbol{c}$ from $\boldsymbol{r}$ is by solving (4) by choosing the noise vector $\boldsymbol{n}$ with the minimum hamming weight.

3.2. Mapping to the Error Correcting Code Problem

We now make the connection to the problem at hand. Consider equation (2) and assume that instead of $\boldsymbol{x}$, we are trying to find out $\boldsymbol{x}^* = \boldsymbol{x}+\boldsymbol{1}$, where $\mathbf{1}$ is the all-one vector, i.e. we are trying to solve $A\boldsymbol{x}^* = \boldsymbol{y} + A\boldsymbol{1} = \boldsymbol{y}^*$. The vector of $\boldsymbol{1}$ is added to convert the congestion localization problem to the ECC problem. Therefore as mentioned earlier, we have, $\Pr\{X_i = 0\} = \rho; \Pr\{X_i = 1\} = 1 - \rho$, but if $\boldsymbol{x}^* = \boldsymbol{x} + \mathbf{1}$, then $\Pr\{X^*_i = 1\} = \rho; \Pr\{X^*_i = 0\} = 1 - \rho$. Let the matrix A and vector $\boldsymbol{y}^*$ be the same as the parity check matrix H and vector $\boldsymbol{d}$ respectively as defined in (4). The elements of $\boldsymbol{x}^*$ are *iid* and have the same distribution as noise $\boldsymbol{n}$ defined above. In the problem of decoding of ECC over BSC as shown in Fig. 4, suppose the transmitted word is $\boldsymbol{x}^*$ and the noise is also $\boldsymbol{x}^*$, the received word is all zero-vector $\mathbf{0}$. Note that we are dealing with non-linear ECC, since H $\boldsymbol{x}^*(A\boldsymbol{x}^*) \neq 0$. Hence, if we could solve $A\boldsymbol{x}^* = \boldsymbol{y}^*$, we can also solve $H\boldsymbol{n}=\boldsymbol{d}$, and be able to decode the transmitted codeword $\boldsymbol{c}$. The information capacity of a binary symmetric channel with bit-error probability ρ is [18]

$$C = 1 - h(\rho) \qquad (5)$$

where $h(x) = -x\log_2 x - (1-x)\log_2(1-x)$ is the entropy function. Shannon's second channel coding theorem implies that [18], unless the rate $R \leq C$, it is not possible to determine the codeword $\boldsymbol{c}$ from

r with probability of error tending to zero asymptotically (i.e. when the length of ***c***, ***r*** tends to infinity). From (5), we have

$$R \le 1 - h(\rho) \qquad (6)$$

For a parity-check matrix H (i.e., A in the congestion localization problem), *1-R* is the ratio of number of rows to the number of columns in H where R is the rate of the code. Thus, (6) gives a lower bound on the ratio of the number of independent observations and the number of network nodes to determine the state of every network node with zero probability of error (asymptotically). Independently, in [19], the authors have shown that *1-R* is the minimum average number of probes per edge and it is lower bounded by $h(\rho)$. The results of the following theorems are also applicable in the congestion localization problem and aids in defining the measure of scalability.

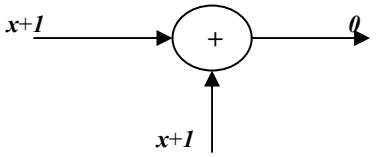

Fig. 4. Channel model for congestion localization

Theorem 1[20]: *For a binary code with parity check matrix A_{LXN} that has a constant row weight d and rate R over a memoryless binary-input symmetric output channel with crossover probability ρ, the necessary condition for reliable communication is*

$$1 - R \ge \frac{1 - C}{h(\rho_d)} \qquad (7)$$

where $\rho_d = \frac{1}{2}\left(1 - (1-2\rho)^d\right)$. From (5) equation (7) reduces to

$$1 - R \ge \frac{h(\rho)}{h(\rho_d)} \qquad (8)$$

The following theorem provides a lower bound on the observation needed for a given matrix A and fixed ρ. In the congestion localization problem, ρ is the probability that a node is congested.

Theorem 2[20]: *For a binary code with parity check matrix A_{LXN} and rate R over a memoryless binary-input symmetric output channel with crossover probability ρ, the necessary condition for reliable communication is*

$$1 - R \ge \frac{1 - C}{\sum_d p_d h(\rho_d)} \qquad (9)$$

where it is assumed without loss of generality where all $(1-R)N$ rows of A are linearly independent and has the property that a p_d fraction of first $(1-R)N$ rows has weight d. From (5), equation (9) reduces to

$$1 - R \ge \frac{h(\rho)}{\sum_d p_d h(\rho_d)} \qquad (10)$$

where $p_d = \frac{k}{n'}$; k nodes with degree d and is the total number of check nodes. Steps for congestion localization is shown below.

1) *Generate the topology G=(V, E)*
2) *Find the shortest paths $\forall v_i \in V$ to $v_j; j \ne i$*
3) *Select the paths of G such that maximum nodes $|V'|$ are covered such that $|V'| \le |V|$ while minimizing the number of probe paths using greedy heuristics.*
4) *Using $1 - R \ge \frac{h(\rho)}{\sum_d p_d h(\rho_d)}$ compute the maximum percentage of congested nodes that can be localized with zero detection error.*

4. SCALABILITY RESULTS

4.1. Scalability Analysis

In this section we provide scalability analysis from the network management point of view along with simulation results. From the theory of linear codes, n is the length of the binary code given the solution set $\boldsymbol{x}$ to the parity-check equation $H\boldsymbol{x}^T = 0^T$, where H is the parity-check matrix. The equation $H\boldsymbol{x}^T = 0^T$ can be written in terms of the Tanner graph where each variable node corresponds to one bit of the codeword, and each check node corresponds to one parity-check equation. The construction of the bipartite graph is such that each non-zero entry in the H (one-one mapping) will have an edge from the variable side to the check side. In the localization problem, the minimum set of probing paths forms the check side and the network nodes in these paths form the variable side as shown in Fig.1. If n' are the total number of nodes on the variable side, then $n' \le n$, where n denotes the total nodes in the physical IP topology. From the bipartite graph construction we have,

$$1 - R = \frac{\#\ of\ observations}{\#of\ network\ nodes} \qquad (11)$$

Therefore from (11), we have

$$1 - R = \frac{\#\ of\ observations}{\#of\ network\ nodes} \ge \frac{h(\rho)}{\sum_d p_d h(\rho_d)} \qquad (12)$$

Equation (12) implies where $c = \frac{h(\rho)}{\sum_d p_d h(\rho_d)}$ is a constant and $0 \le c \le 1$. That is, the number of measurements needed to identify all congested nodes grows linearly with the size of the network. Intuitively, as shown in Fig.7(a), given the ratio $\frac{m}{n'} = 1 - R = 0.5$, where m is the number of observations, the maximum fraction of congested nodes (ρ^*) that can be detected without error is *0.11* from equation (5).

Since, the routing matrix A does not have equal row weights, the ρ that can provide zero detection error is less than ρ^* which is shown as the achievable region and is also confirmed by simulation. Hence if we fix the set of observations (probing paths), the pdf of the check side p_d and the number of nodes of the graph n' are known, and hence ρ can be obtained. On the other hand, if we know the value of the percentage of congested nodes, ρ apriori then the growth rate of the number of measurements with respect to the network size that can result in zero detection error can be determined. Therefore the bound in (12) provides us the measure of scalability.

4.2. Simulation Results

We use the message-passing algorithm [21] to obtain the status of the network nodes using minimum number of independent measurements. The message-passing algorithm approximates the maximum likelihood probability (ML) by constructing a bipartite graph that consists of the variable and the check side as shown in Fig. 1. Simulations were performed on a random graph of various sizes, i.e., n=500, 1000, 2000 and 5000. This is similar to increasing the code length in the ECC framework. The ratio of the observations to the number of nodes in the bipartite graph was chosen to be 0.5 in each case and the prior ρ was varied. The paths were chosen according to shortest path routing algorithm. The observation side degree distribution corresponds to the hop count distribution observed on the Internet which is gamma distributed and is shown in Fig. 5(b). In particular, simulations were averaged over 1000 patterns and the detection error was computed. The bound in (6) ensures that up to a prior of *11%* of congested nodes can be detected for $m/n' = 1 - R = 0.5$ with zero error provided the network is large and the *H* matrix is well designed. The variable side distribution is shown in Fig. 6(a). Hence from equation (10) the bound is reduced to *10.1%* due to the scaling factor $\sum_d p_d h(\rho_d)$.

From simulation we could detect all errors up to *4%* of the prior. Fig. 6(a) depicts the following, where $\rho^* = 10.1\%$ and the detection error shows a steep fall after *4%* of the prior. This implies that below the lower bound ($\rho^* = 10.1\%$), we can guarantee that all congested nodes can be localized with zero detection error.

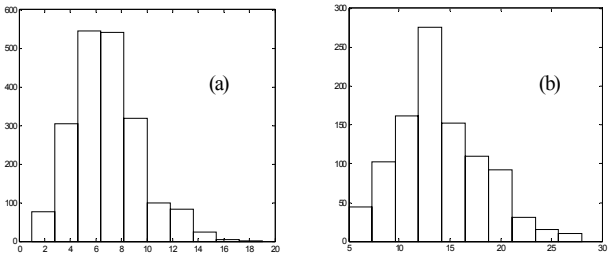

Fig. 5. Degree distribution on the (a) variable side (b) check side for a network of size 2000 nodes and the number of observations=1000.

In Fig. 7(b), we have shown the effect of adding measurement noise in the detection problem. The noise values for each measurement were assumed to be independent of each other. We note that the detection error increases with the amount of noise, as expected. It can also be seen that the unlike in the noiseless case, the detection error decreases very slowly with increasing prior. We believe that longer probe lengths are responsible for this phenomenon.

The simulation results are far from the theoretical bound for the following reasons. The result in (12) is valid asymptotically, i.e. when the code length (i.e., network size) tends to infinity. When the code length is finite as opposed to (6), which is valid asymptotically, the following sphere-packing lower bound for BSC with the cross over probability ρ holds [22].

Theorem 3[22]: *An (n, k) code on the BSC has probability of error lower-bounded by*

$$P(n,k,\rho) = \sum_{i=r+1}^{n} \binom{n}{i} \rho^i (1-\rho)^{n-i} - \rho^{r+1}(1-\rho)^{n-r-1}\left(2^{n-k} - \sum_{i=0}^{r}\binom{n}{i}\right) \quad (13)$$

where r is defined as the maximum integer such that $\sum_{i=0}^{r}\binom{n}{i} \le 2^{n-k}$

and $P(n,k,\rho)$ *is the pattern error rate which is defined as the ratio of number patterns in error (single/multiple bit errors) to the total number of patterns.*

Theorem 3 provides a lower-bound for finite length codes where the rate $R=k/n$ and $q = \frac{\#\ of\ observations}{\#\ of\ network\ nodes} = 1 - k/n$.

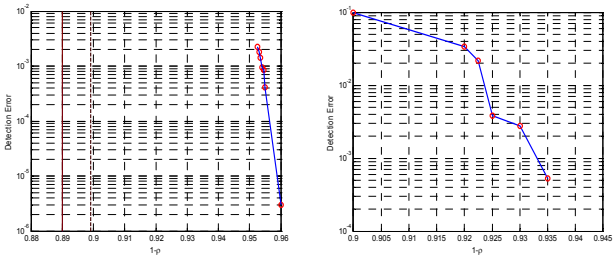

Fig. 6. (a) Detection Error vs. prior for a network of 2000 nodes and 1000 observations. (b) Detection Error vs. prior for a network of 2000 nodes and 1986 observations.

Furthermore, the message-passing algorithm assumes that the bipartite graph is cycle-free, which is the optimal case, which is not possible in this setting. Whereas if the graph has cycles in the case of congestion localization, the message passing algorithm no longer gives the exact ML result, but it is an approximate solution and hence the performance deviates from the theoretical bound. From the graph we can see that we are about 6.1% away from the bound due to the reasons explained above. Fig. 6(b) depicts the detection error for a network of size *n=2000* where all the nodes are on the bipartite graph and observations *m=1986*. Scalability results are shown in Fig. **7**(a) where the y-axis represents the minimum number of probes that are required to detect the congested nodes/links and the x-axis represents the maximum fraction of prior that ensures zero detection error. The figure demonstrates linear relation between the observations and the prior and hence the proposed approach is scalable. If we could tolerate a finite detection error then the number of observations can be further reduced.

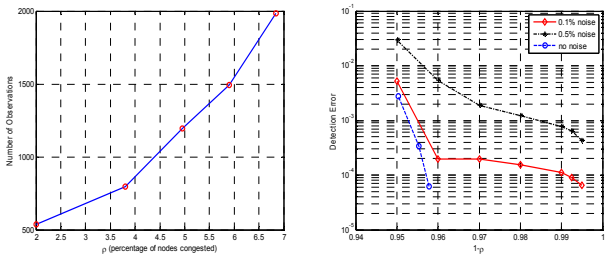

Fig. 7. (a) Scalability analysis: Fraction of congested nodes vs. the number of observations for a network of size 2000, (b) Detection error vs. prior in the presence of measurement noise up to 0.5%.

4.3. Computational and Implementation Cost

The computational complexity of the message-passing algorithm is proportional to $O(n \log n)$ where *n* is the block length of the LDPC code. In the congestion localization problem, the computational complexity depends on the number of network nodes in the bipartite graph. As mentioned in section 3, measurements are performed using probe packets that have a payload of one bit. Hence the size of each probe packet is 41 bytes including the header. TCP traffic consists of packets of three sizes: 40 byte packets (the minimum packet size for TCP) that carry TCP

acknowledgments without any payload; 1500 byte packets which is the maximum Ethernet payload size from TCP implementations that use path Maximum Transmission Unit (MTU) discovery and 552 byte/576 byte packets from TCP implementations that do not use path MTU discovery. Hence even if every packet were considered as a probe, the additional bit payload accounts for 2.73% of the maximum packet size. If probe packets were designated for network monitoring only, the traffic volume due to these packets would be much smaller than the total flows due to actual traffic and hence does not burden the network. This approach is practically realizable on networks where priority queuing algorithms can be implemented. This is because, the probe packets needs to be sent to the next node when the node/link is congested which, has to be performed in real-time. We also assume that all the nodes in the path are functioning and have the capability to flip the bit in the probe packet in case the node is not congested.

5. CONCLUSIONS AND FUTURE WORK

We investigate the scalability of end-to-end measurement-based network monitoring in the context of congestion localization. We formulate the congestion localization problem as decoding of linear error control codes over a BSC and provide theoretical lower bounds. Scalability is defined as the ratio of the number of independent observations to the number of nodes in the bipartite graph. We show that the proposed approach is scalable, i.e., it provides a linear growth rate of number of measurements with respect to the network size. Simulations are performed on network of various sizes to verify the scalability result. The inference is performed using message-passing that has a low computational complexity by forming a bipartite graph from the probed paths and the nodes on those paths. In the current approach, only one additional bit of payload is needed in the probe packet and hence the traffic volume generated due to probe packets does not burden the network. In this work, we assume that each node gets congested independently and this assumption can be relaxed to consider correlated behavior as an extension. We also plan to investigate the scalability when the prior probability is unknown and obtain the detection error. Intuitively, if we could design a *H(A)* matrix that can achieve the lower bound by reducing the cycles in the bipartite graph and then the probing paths are designed, the number of probes can be reduced significantly. Furthermore, our method can be extended to the scenario where the end-hosts form an overlay network that assumes the knowledge of the underlying IP topology as in [23] and also in detecting security holes in the network, which might cause denial-of service attacks.

11. REFERENCES

[1] S. Iyer, S. Bhattacharyya, N. Taft, and C. Diot, "An approach to alleviate link overload as observed on an IP backbone," *INFOCOM*, 2003.

[2] A. Habib, M. Khan, and B. Bhargava, "Edge-to-edge measurement-based distributed network monitoring," *Computer Networks*, vol. 44, pp. 211-233, Feb 2004.

[3] Y. Breitbart, C.-Y. Chan, M. Garofalakis, R. Rastogi, and A. Silberschatz, "Efficiently Monitoring Bandwidth and Latency in IP Networks," *Proceedings of IEEE INFOCOM*, 2000.

[4] Z. Cataltepe and P. Moghe, "Characterizing Nature and Location of Congestion on the Public Internet," *IEEE Symposium on Computers and Communication, Kemer, Antalya, Turkey*, 2003.

[5] D. Rubenstein, J. Kurose, and D. Towsley, "Detecting Shared Congestion of Flows Via End-to-end Measurement," *Proceedings of ACM SIGMETRICS'00*, June 2000.

[6] D. Katabi, I. Bazzi, and X. Yang, "A Passive Approach for Detecting Shared Bottlenecks," *IEEE International Conference on Computer Communications and Networks, ICCCN 2001, Arizona*, 2001.

[7] A. Tachibana, S. Ano, T. Hasegawa, M.Tsuru, and Y. Oie, "Empirical Study on Locating Congested Segments over the Internet Based on Multiple End-to-End Path Measurements," *The 2005 Symposium on Applications and the Internet*, pp. 342 - 351, 31-04 Jan. 2005.

[8] N. Hu, L. E. Li, Z. M. Mao, P. Steenkiste, and J. Wang, "Locating Internet Bottlenecks: Algorithms, Measurements, and Implications.," *SIGCOMM*, 2004.

[9] A. Akella, S. Seshan, and A. Shaikh, "An Empirical Evaluation of WideArea Internet Bottlenecks," *In Proceedings of ACM SIGCOMM Internet Measurement Conference (IMC), Miami, FL*, October 2003.

[10] C. Tang, P. K. McKinley, and J. Shapiro, "Collaborative Path Selection for Topology-Aware Overlay Path Monitoring in the Presence of Selfish Agents," *Technical Report MSU-CSE-04-42*, October 2004.

[11] C. Ji and A. Elwalid, "Measurement-Based Network Monitoring: Achievable Performance and Scalability," *IEEE Journal of Selected Areas of Communication:Special Issue on Recent Advances in Fundamentals of Network Management*, vol. 20, pp. 714-725, May 2002.

[12] M. Coates, A. Hero, R. Nowak, and B. Yu, "Internet Tomography," *IEEE Signal Processing Magazine*, vol. 19, pp. 47-65, May 2002.

[13] I. Rish, M. Brodie, and S. Ma, "Accuracy vs. efficiency trade-offs in probabilistic diagnosis," *Eighteenth national conference on Artificial Intelligence*, pp. 560-566, 2002.

[14] S. Geman and K. Kochanek, "Dynamic Programming and graphical representation of error-correcting codes," *IEEE Transactions on Information Theory*, vol. 47, pp. 549-568, 2001.

[15] Y. Mao, F. R. Kschischang, B. Li, and S. Pasupathy, "A Factor Graph Approach to Link Loss Monitoring in Wireless Sensor Networks," *IEEE JSAC*, vol. 23, pp. 820-829, April 2005.

[16] M. Steinder and A. S. Sethi, "Probabilistic fault localization in communication systems using belief networks," *IEEE Transactions on Networking*, vol. 5, pp. 809-822, October 2004.

[17] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu, "Internet Tomography: Recent Development," *Statistical Science*, 2003.

[18] T. M. Cover and J. A. Thomas, *Elements of Information Theory*: Wiley, August 1991.

[19] Y. Wen, V. W. S. Chan, and L. Zheng, "Efficient Fault Diagnosis Algorithms for All-Optical WDM Networks," *IEEE/OSA Journal of Lightwave Technology Special Issue on Optical Networks*, October 2005.

[20] D. Burshtein, M. Krivelevich, S. Litsyn, and G. Miller, "Upper Bounds on the Rate of LDPC Codes," *IEEE Transactions on Information Theory*, vol. 48, pp. 2437-2449, Sept 2002.

[21] T. Richardson and R. Urbanke, "The Capacity of Low-Density Parity Check Codes under Message-Passing Decoding," *IEEE Transactions on Information Theory*, vol. 47, pp. 599-618, Feb 2001.

[22] R. G. Gallager, "Low Density Parity Check Codes," *MIT Press, Cambridge, MA*, 1963.

[23] Y. Chen, D. Bindel, H. Song, and R. H. Katz, "An Algebraic Approach to Practical and Scalable Overlay Network Monitoring," *ACM SIGCOMM*, 2004.