The inference of link loss rates with internal monitors

Haibo Su, Wentao Chen, Shijun Lin, Depeng Jin, Lieguang Zeng

State Key Laboratory on Microwave and Digital Communications
Tsinghua National Laboratory for Information Science and Technology
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
Email: suhb02@gmail.com

Abstract—Network tomography has been widely used recently as an method to infer the network internal link-level characteristics by end-to-end measurement. In this paper, we consider the problem of estimating link loss rates using network tomography. The existing methods make the inference based on the whole tree of network, which is very complex for large scale network. To overcome this limitation, we propose a low complexity inference approach named LCIA. In the LCIA, we deploy monitors at internal nodes to reduce the complexity of inferring the link loss rates. It mainly consists of two steps. The first step is to deploy monitors at specific internal nodes to divide the original tree into several sub-trees with minimum depth. The second step is to infer the link loss rates of sub-trees by a new estimator which is an explicit function of loss measurements. The LCIA has the following features. First, it greatly reduces the inference complexity as the inference on the sub-trees is much simpler. Second, it improves the accuracy of the estimated results since the variance of loss estimator on sub-trees with lower depth is smaller than that on the original tree. The analytical and simulation results demonstrate that the LCIA outperforms the existing methods both on computation complexity and inference accuracy.

Index Terms—Network tomography, Link loss rates, Internal monitors.

I. INTRODUCTION

In order to monitor and manage the network, it is essential to identify a large number of network performance parameters such as link loss rate [2], [5], [6], [9], delay distribution [11] and logic topology [12]. As Internet grows fast in size and diversity, today it has evolved into a large complex heterogeneous and unregulated system, which makes the direct measurement of network characteristics infeasible. To deal with this challenging problem, network tomography [10] is considered as an effective approach to infer network internal performance parameters by end-to-end measurement. This new concept is first presented by Vradi [1] in 1996, and the term tomography is often used in medical science and computed tomography (CT).

The basic principle of network tomography is to extract the hidden information from active or passive end-to-end measurement data by statistic inference. Probing packets are sent, either by multicast or unicast, from a root node to a number of destination leaf nodes attached to the network with tree topology. The observations of the receivers that share the same parent or ancestor have relatively strong correlation, because packets have similar experience on the shared paths. The internal network properties can be inferred by exploiting the inherent correlation in measurement results.

Here, we focus on the inference of link loss rates, which is also called loss tomography. In the past few years, a number of approaches, including maximum-likelihood estimator (MLE) method and expectation maximization (EM) algorithm, have been proposed to estimate the loss parameter. Caceres et al. are the pioneer to develop MLE method for link loss rates by multicast probing packets [2], [3], [4]. It needs to solve a set of high order polynomial equations which show the correlation of loss rates between the internal node and its children. Coates and Nowak proposed a back-to-back packet pair unicast measurement scheme, and used EM algorithm to estimate the correlation between the packet pairs [6]. It executes the iterative procedure to search for a solution in a multi-dimensional space. These existing methods all infer the link loss rates based on the whole tree of original network. With the growth of network’s scale, both of the two methods are very complex and time-consuming, and this will restrict the techniques to be used in practice.

To overcome these shortcomings, in this paper we propose a low complexity inference approach LCIA by deploying monitors at internal nodes to infer the link loss rates. It mainly consists of two steps. The first step is to deploy monitors at specific internal nodes to divide the original tree into several sub-trees with minimum depth. In this step, we present an optimal algorithm of how to deploy the monitors to make the number of monitors minimized if the maximum depth of the sub-trees is limited. The second step is to infer the link loss rates of the divided sub-trees by a new estimator which is an explicit function of loss observations. Since the inference on the sub-trees is much simpler than that on the original tree and the explicit estimator only requires arithmetic calculation, the LCIA greatly reduces the computation complexity. Moreover, the variance of loss estimator on sub-trees with lower depth...
is smaller than that on the original tree, so it improves the accuracy of the estimated results. The analytical and simulation results show that the LCIA outperforms the existing methods both on computation complexity and inference accuracy.

The remainder of the paper is organized as follows. In Section II, the mathematical link loss model for multicast tree-like network is introduced. In Section III, we give an explicit estimator for the link loss rate. In Section IV, an algorithm is presented to solve the problem of how to deploy the monitors at internal nodes to divide the original tree into several subtrees. In Section V, we propose the approach LCIA to infer the link loss rates, and analyze its computation complexity and estimation accuracy. In Section VI, the accuracy of LCIA is validated through simulation. Finally, we conclude the paper in Section VII.

II. LOSS MODEL FORMULATION

The network with tree topology is represented by a logic multicast tree model. This model is denoted by $T = (V, L)$, where $V$ is the node set and $L$ is the link set [2]. The source is the root node 0 which sends probing packets to receivers $R$ where $R \subset V$. Apart from the root node, each node $k \in V$ has a unique parent $f(k)$. In addition, we define $j = f^n(k) = f(f^{n-1}(k))$ with $f^{(0)}(k) = k$ if $k$ is descended from $j$, which is also expressed as $k < j$ or $j > k$. To be simple, we denote the link $(f(k), j) \in L$ that terminates at node $k$ as link $k$. As to each non-leaf node $k$, it has a set of $d(k)$ child nodes $D(k) = \{d_i|1 \leq i \leq d(k)\}$. As in graph theory, we define $T(k) = (V(k), L(k))$ as the subtree of $T$ rooted at node $k$. The receiver node set contained in $T(k)$ is denoted as $R(k) = R \cap V(k)$.

We assume that link losses are described by independent Bernoulli losses, and losses for different links and different packets are independent. For each link $k \in L$, the probability that a probing packet is lost on this link is $\theta_k$. Therefore the progress that each given packet propagates down the multicast tree can be modeled by an independent statistical process $X = \{X_k\}_{k \in V}$ where the value of each $X_k$ falls in $\{0, 1\}$. $X_k$ is equal to one when a probing packet successfully reaches node $k$ and zero if it dose not. Then we easily get $X_0 = 1$ since the probing packets are sent by the root node. As to any non-root node $k$, if $X_k = 0$, then $X_j = 0$ for all the children $j$ of $k$ (i.e., $j \in D(k)$). Otherwise, if $X_k = 1$, according to the definition of $\theta_k$, we have $P[X_j = 0|X_k = 1] = \theta_j$ and $P[X_j = 1|X_k = 1] = 1 - \theta_j$. If we dispatch N probing packets, then $X^{(k)}_i(1 \leq i \leq N)$ denotes the loss measurement of node $k$ for the $i$th probing packet, and the 0-1 sequence vector $\{X^{(k)}_i\}_{k \in V}$ is one outcome of the statistical process $X = \{X_k\}_{k \in V}$. The task of loss tomography is to estimate the value of $\theta_k$ from the loss observation $X(i) = \{X^{(k)}_i\}_{k \in R}$ at the leaf nodes which is the outcome of statistical process $X_R = \{X_k\}_{k \in R}$.

III. THE EXPLICIT ESTIMATOR OF LOSS RATES

For each internal node $k$ apart from root and leaf nodes, a two level tree can be constructed which consists of an input virtual link $V_k$, connecting the root node 0 to $k$, and a number of output virtual links, each for a subtree descended from node $k$ [8]. An input virtual link $V_k$ is, in fact, the path composing of links through which the probes are sent from root node to node $k$. The loss rate of path $V_k$ is defined as the probability that a probe originating at the source failed to reach node $k$, which is expressed as

$$p_k = 1 - (1 - \theta_k) \prod_{j < k} (1 - \theta_j)$$

and $p_0 = 0$. Therefore for each non-root node $k$, the loss rate of corresponding link $k$ can be denoted by the loss rates of input virtual links

$$\theta_k = 1 - (1 - p_k)/(1 - p_{f(k)}).$$

Accordingly, if we firstly estimate the path loss rates by the loss observations at the leaf nodes, then the link loss rate $\theta_k$ can be identified using Eq. (2).

Each output virtual link of internal node $k$ represents a subtree descended from node $k$. The corresponding subtree set can be expressed as $\{T(d_i^k) : 1 \leq i \leq d(k)\}$ where $d(k)$ is the number of children of node $k$. The set of receiver nodes contained in $T(d_i^k)$ is $R(d_i^k)$ which is a subset of $R(k)$. In order to combine the observations at the leaf nodes attached to subtree $T(k)$, we define a new random variable $T_k$:

$$T_k = \left\{ \begin{array}{ll} 1, & \exists i, i \in R(k), X_i = 1 \\ 0, & \forall i, i \in R(k), X_i = 0 \end{array} \right.$$ (3)

where $X_i$ denotes the loss observation at leaf node $i$. At the same time, the 0-1 sequence $\{T^{(k)}_1, T^{(k)}_2, \cdots, T^{(k)}_{d(k)}\}$ represents the loss observation of subtree $T(k)$ if N probing packets are dispatched. In addition, the loss rate of subtree $T(k)$ is defined as the probability that a probing packet fails to reach each of its receivers $R(k)$, which is denoted as $L_k = P[T_k = 0]$.

For instance, Fig. 1 shows how to construct a two level tree for internal node 2 of a multicast tree shown in Fig. 1a. In this multicast tree there are 7 internal nodes, i.e. 1, 2, 3, 4, 5, 6 and 7. For each internal node, a two level tree with one input virtual link and a number of output virtual links can be constructed. For example, the constructed two-level tree for node 2 is shown in Fig. 1b. It has one input virtual link which is the path from root node 0 to node 2 and two output virtual links which represent the subtrees $T_1$ and $T_2$ respectively.

If internal node $k$ has $d(k)$ children, then its corresponding output virtual links are $T(d^0_k), T(d^2_k), \cdots, T(d^{d(k)}_k)$. For any positive integer $m \leq d(k)$, We define the probability

$$r_{b_1b_2\cdots b_m}(k) = P\{T^{(k)}_{d^1_k} = b_1; T^{(k)}_{d^2_k} = b_2; \cdots; T^{(k)}_{d^m_k} = b_m\}$$ (4)

and

$$r^{(i)}_{b_i}(k) = P\{T^{(k)}_{d^i_k} = b_i\}$$ (5)

where $b_i \in \{0, 1\}(1 \leq i \leq m)$. The corresponding estimator can be easily achieved by the loss observations at the leaf nodes

$$r_{b_1b_2\cdots b_m}(k) = n\{T^{(1)}_{d^1_k} = b_1; T^{(2)}_{d^2_k} = b_2; \cdots; T^{(m)}_{d^m_k} = b_m\}/N$$ (6)
is the number of outcomes which satisfy the condition

\[
\text{Then Eq. (8) is proved. From Eq. (6) and Eq. (7), we know}
\]

**Theorem 1:** For non-root node \( k \in V/R \)

\[
p_k = 1 - \widehat{p}_k = 1 - \left[ \prod_{i=1}^{d(k)} \frac{r^{(i)}_b(k)}{r^{(i)}_{11...1}(k)} \right]^{1/(d(k)-1)}
\]

(8)

where \( r^{(i)}_{11...1}(k) \) denotes the probability \( r^{(i)}_{11...1}(k) = P\{T_{d_{k}} = 1; T_{d_{k}'} = 1; \cdots T_{d_{k}^{(m)}} = 1\}. \)

**Proof:** According to [13, Th. 5], we have

\[
\frac{d(k)}{r^{(i)}_b(k)} \left[ \prod_{i=1}^{d(k)} \frac{r^{(i)}_b(k)}{r^{(i)}_{11...1}(k)} \right]^{1/(d(k)-1)}
\]

(9)

Then Eq. (8) is proved. From Eq. (6) and Eq. (7), we know that the estimators of \( r^{(i)}_{b_1b_2...b_m}(k) \) and \( r^{(i)}_b(k) \) are both explicit functions of the loss observations at the leaf nodes \( R(k) \). In the following we derive the estimator of link loss rate \( \theta_k \).

**Theorem 2:** For non-root node \( k \in V/\{R\} \), the estimator of link loss rate \( \theta_k \)

\[
\hat{\theta}_k = 1 - \widehat{\theta}_k = 1 - \frac{\widehat{p}_k}{\overline{p}_k} = 1 - (1 - \widehat{p}_k)(1 - \overline{p}_k)
\]

(10)

is a consistent estimator, where \( \overline{\theta}_k = 1 - \theta_k \) and \( \overline{p}_k = 1 - p_k \).

**IV. DIVIDE THE ORIGINAL NETWORK WITH INTERNAL MONITORS**

**A. Dependence of the inference accuracy and complexity on the depth of the network**

The tree depth is the maximum level of any leaf node in the network with tree topology. It is an important factor that influences the accuracy and complexity of the inference methods. We have several conclusions which describe their relationships:

1) The variance of loss rate estimator increases with tree depth of the network.
2) The complexity of inference method increases with tree depth of the network.

Here we give some explanations to these two conclusions. The conclusion (1) can be obtained intuitively. The cumulative error per link in estimation is increased with the tree depth. As to the MLE method, the variance of MLE estimator is, to first order in the value of loss rate, independent of topology of the network (see the proof of [2, Th.5]). In this kind of method, the variance of loss rate estimator increases with tree depth, roughly linearly (see the proof of [2, Th.5]). The increase of estimator variance will make the estimate results of loss rates more inaccurate. The conclusion (2) is also apparent. The total measurement data collected at all the leaf nodes decreases with the tree depth. Also the loss estimator for the smaller tree depth network will become simpler. These two factors will lead to the decrease of the computation complexity of the methods.

**B. The division of the original tree**

The measurement results for the existing methods are only the loss observations at the leaf receivers. We don’t know the details of loss events occurred at the internal nodes. If we deploy an monitor at one internal node, the loss events at this node can also be observed.

The next question is how to deploy monitors at the internal nodes to divide the original tree. We use \( M \) to denote the number of deployed monitors. If we assume that the maximum depth of the divided subtrees is limited, then the question is to
how to deploy the monitors to make the number of monitors $M$ minimized.

As to every leaf node $k$, we define $H(k)$ as the distance from node $k$ to root node. The depth of network $H$ is the maximum value of $H(k)$, which can be expressed as $H = \max_{k \in R} H(i)$.

We use $h_{\text{max}}$ to denote the limited value of maximum depth of sub-trees. Then we hope that the depth of every divided sub-tree is close to the upper value $h_{\text{max}}$ in order to reduce the number of monitors. Here we propose an tree division algorithm named TDA to address this problem:

Algorithm TDA:

1) Find the sub-tree with the largest depth, named LST, among all the divided sub-trees. At the beginning, the found LST is the original tree $T$. If the original tree $T$ has been divided into several sub-trees by the deployed monitors, we choose one sub-tree with the biggest depth comparing to other sub-trees. If the depth of LST is not larger than $h_{\text{max}}$, then quit. Go to Step2.

2) For the found sub-tree LST, choose one leaf node $k$ such that $H(k) = \max_{j \in R_{\text{LST}}} H(j)$ where $R_{\text{LST}}$ is the leaf node set of LST. Then the distance of node $k$ to the root node of LST, $H(k)$, is not less than that of any other leaf node.

3) Find out the ancestor node of selected node $k$, $f^{h_{\text{max}}}(k)$, recursively by $f^n(k) = f(f^{n-1}(k))$. As the depth of LST is larger than $h_{\text{max}}$, the ancestor node $f^{h_{\text{max}}}(k)$ exists.

4) There are $h_{\text{max}}$ links connecting node $k$ and its ancestor node $f^{h_{\text{max}}}(k)$. If we deploy one monitor at internal node $f^{h_{\text{max}}}(k)$, then one divided smaller tree is the sub-tree rooted at node $f^{h_{\text{max}}}(k)$. Apparently its depth is equal to $h_{\text{max}}$. Then go to Step1.

We use the algorithm TDA to divide the original tree shown in Fig. 2a. The depth of this original tree is four and we assume that the maximum depth of divided sub-trees $h_{\text{max}} = 2$. Using the TDA, we deploy two monitors at internal node 2 and 3 to divide the original tree into three sub-trees shown in Fig. 2c. The maximum depth of these three divided sub-trees $h_{\text{max}} = 2$.

In the following part, we prove that the algorithm TDA is optimal.

Theorem 3: If the limited value of maximum depth of sub-trees is $h_{\text{max}}$, $M_{a_{\text{lg}}}$ denotes the number of monitors required by TDA , and $M_{\text{min}}$ denote the minimum number of monitors required. Then $M_{a_{\text{lg}}} = M_{\text{min}}$.

Proof: if $k = f^n(j)$ or $j = f^n(k)$ for some integer $n > 0$, we use $d(j,k) = n$ to denote the distance of nodes $j$ and $k$. If not, we define $j \land k$ as the closest common ancestor of nodes $j$ and $k$. At this time, the distance of node $j$ and $k$ is defined as $d(j,k) = \max\{d(j,j \land k), d(k,j \land k)\}$. Take the tree in Fig. 2a as an example, we have $d(1,4) = 2, d(4,12) = 3$ according to the definition of $d(j,k)$. In the step 2 of TDA, one leaf node with the biggest distance to root node comparing to the other leaf nodes in LST will be chosen. We use $v_i$ to denote the chosen node in the $i$th time. Since there are $M_{a_{\text{lg}}}$ monitors deployed in TDA, so the set of all the chosen nodes is $\{v_1, v_2, \ldots, v_{M_{a_{\text{lg}}}}\}$. For example, the chosen node set of the tree in Fig. 2a is $\{8, 12\}$. In TDA, the depth of every new divided sub-tree is equal to $h_{\text{max}}$ except for the last sub-tree. Then for any two nodes $v_i$ and $v_j (i \neq j)$, we have $d(v_i, v_j) \geq h_{\text{max}}$. Equation $d(v_i, v_j) = h_{\text{max}}$ exists only when $v_i \triangleq v_j$ or $v_j \triangleq v_i$. In the following part we want to prove that we at least deploy $M_{a_{\text{lg}}}$ monitors to cover these $M_{a_{\text{lg}}}$ chosen nodes no matter what kind of other deployment schemes we adopt. A monitor at one internal node can cover the nodes whose distance to this node is not larger than $h_{\text{max}}$. If $d(v_i, v_j) < h_{\text{max}}$, two monitors are required to cover them. If $d(v_i, v_j) = h_{\text{max}}$, we have $v_i \triangleq v_j$ or $v_j \triangleq v_i$. In general, we can assume that $v_i \triangleq v_j$. It appears that we can only deploy one monitor at node $v_j$ to cover both $v_i$ and $v_j$. However, if the monitor is deployed at node $v_j$, it, in fact, can not cover the node $v_j$, because at this time $v_j$ is the leaf node of another sub-tree and we need another monitor to cover it. So we also need two monitors to cover $v_i$ and $v_j$ even if $d(v_i, v_j) = h_{\text{max}}$. This means that one monitor can only cover one chosen node no matter what place it is deployed. So we at least need $M_{a_{\text{lg}}}$ monitors to cover these $M_{a_{\text{lg}}}$ chosen nodes. This means that $M_{a_{\text{lg}}} \leq M_{\text{min}}$. At the same time, $M_{a_{\text{lg}}} \geq M_{\text{min}}$ also exists because $M_{\text{min}}$ denotes the minimum number of monitors required in theory. so we derive that $M_{a_{\text{lg}}} = M_{\text{min}}$. This demonstrates that our proposed algorithm for the deployment problem is optimal.

For different value of $h_{\text{max}}$, we use the algorithm TDA to divide the original tree and get the minimum number of monitors $M_{\text{min}}$ required. For a given network, the relationship between $M_{\text{min}}$ and $h_{\text{max}}$ can be obtained by TDA, and it can be denoted by $M_{\text{min}} = F(h_{\text{max}})$. The Fig. 3 describes the
relationship of these two parameters for a set of complete binary trees with original tree depth $H = 7, H = 6$ and $H = 5$.

V. THE LOW COMPLEXITY APPROACH WITH INTERNAL MONITORS

In this part, we propose the low complexity approach LCIA to infer the link loss rates. It mainly consists of two steps.

In the first step, we deploy monitors at internal nodes to divide the original tree into smaller sub-trees. It is difficult to give a general solution to the problem of how to deploy the monitors. We assumed that the maximum depth of divided sub-trees is limited, then the deployment problem can be simplified. As to the simplified problem, we have proposed an optimal algorithm TDA to solve it. Therefore the original tree can be divided into several smaller sub-trees using the deployed monitors.

As to the existing methods to infer the link loss rates, the measurement results available are only the loss observations at the leaf nodes of the network. The loss events happened at the internal nodes is unknown to us at all. However, if we deploy one monitor at one internal node, then whether the probing packets arrived at this node can be observed. Therefore all the loss events happened at internal nodes with monitors are available. An internal node with one monitor is a leaf node of one divided sub-tree, then it can be seemed as a receiver of this sub-tree. Moreover, an internal node with one monitor is also the root node of another divided sub-tree, then it can be seemed as the packet sender if we only concern about the packets arrived at this node. So every sub-tree has its associated packet sender and packet receivers.

In the second step, we use the proposed explicit estimators proposed in Eq. (8) and Eq. (10) to identify the link loss rates of each divided sub-tree. As these sub-trees are independent, so the loss rates can be computed in parallel and hence the computation time is reduced. The depth of each sub-tree is smaller than that of the original tree, so the computational complexity is decreased. The estimator proposed in Eq. (10) is an explicit function of the measurement results, so this method only requires simple arithmetic computation. This can greatly reduce the computation complexity comparing to the MLE method when the scale of network is very large. Also this estimator is a consistent estimator of the real link loss rate and has the same asymptotic variance as the MLE.

VI. SIMULATION RESULTS

We use a 16-node network shown in Fig. 4a to validate the effectiveness of the LCIA. In this experiment, probing packets are sent from the root node 0. There are 15 links involved in this test which loss the packets in a fixed rate in sufficient short period that can be denoted as a probing interval (PI) [7].

In each PI, we send 1000 packets from the node 0 based on multicast. The loss rates of each link located in this network are set to be 10%. For each non-root node, the real loss rate of its corresponding link is one minus the proportion between the number of packets which reach this node and that reach its parent node.

The experiment is conducted for 20 probing intervals. We use two methods to estimate the link loss rates. The first one is to make the inference only based on the explicit estimators in Eq. (8) and Eq. (10) without any internal monitor. It only use the loss observations at the leaf nodes to infer the link loss rates. The second one is the LCIA, and we deploy the internal monitors at internal nodes 2 and 3 to divide the original tree into three sub-trees shown in Fig. 4b. Then the loss events happened at internal nodes 2 and 3 can also be observed.

The Fig. 5 describes the experiment results from the simulation which contains 7 subfigures, one for a link and from link 1 to link 7. Each sub-figure in this figure contains three curves, two estimated loss rate curves using the two methods and one actual loss rate curve for the corresponding link.

From the Fig. 5, we can see that the two curves of estimated loss rates vary around the curve of actual loss rates. This shows the explicit estimator of link loss rates converges to the real loss rate. In comparison between the two curves of estimate loss rates, we find that the curve of the second method is closer to the curve of actual loss rates than that of the first method.

![Fig. 3](image1.png)  \[ M_{\text{min}} \text{ decreases with } h_{\text{max}} \text{ for different tree depth} \]

![Fig. 4](image2.png)  \[ \text{Experiment network and its loss rates} \]
and that of the second method in range deploying monitors at internal nodes to infer the link loss error belongs to the range rate and its actual loss rate in each PI, and the value of this can see that the number of errors of the second method in range the number of errors for all the PIs in these two methods. We portrays the histograms which demonstrate the distribution of inference results.

Hence, the simulation results demonstrate that the estimate results of the LCIA fit to the actual loss rates well, and deploying internal monitors can improve the accuracy of the inference results.

**VII. CONCLUSION**

In this paper, we have proposed the approach LCIA by deploying monitors at internal nodes to infer the link loss rates. Using the optimal algorithm TDA, the original tree is divided into several sub-trees with minimum depth by internal monitors. We use an explicit estimator to infer the link loss rates of the sub-trees in parallel. Since the inference of link loss rates on the sub-trees is much simpler than that on the original tree and the explicit estimator only requires arithmetic calculation, the approach LCIA greatly reduces the computation complexity. Moreover, the variance of loss estimator on sub-trees with lower depth is smaller than that on the original tree, so it improves the accuracy of the estimation results. The analytical and simulation results show that the LCIA outperforms the existing methods both on computation complexity and inference accuracy.

**REFERENCES**


