

Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Computer Communications xx (xxxx) 1–15

computer
 communications

www.elsevier.com/locate/comcom

A bottom-up inference of loss rate

Weiping Zhu^{a,*}, Zhi Geng^b^a*School of Computer Science, The University of New South Wales, Canberra ACT2600, Australia*^b*Institute of Mathematical Science, Peking University, Beijing, China*

Received 28 October 2003; revised 5 August 2004; accepted 19 August 2004

Abstract

Loss tomography, as a key component of network tomography, aims to obtain the loss rate of each link in a network by end-to-end measurements. If knowing the loss model of a link, we, in fact, deal with a parametric estimate problem with incomplete data. Maximum likelihood estimates are often used in this situation to identify the unknown parameters in the loss model. Almost all methods proposed so far rely on the iterative approximation to identify the parameters that requires a long execution time. In addition, the parameters identified by those methods may not be the true values of those parameters since the iterative procedure may trap into a local maximum. In this paper, we propose an estimate that is based on the correlation between a link and its sibling brothers to identify the loss rate of the link. The proposed method, instead of using an iterative approach to approximate the maximum, employs a bottom-up approach to identify the loss rates of the links of a network. Comparing to the previous methods, the proposed method is simple and fast because it is an analytical solution.

© 2004 Published by Elsevier B.V.

Keywords: Network tomography; Loss tomography

1. Introduction

Network characteristics, such as loss rate, average delay, available bandwidth, are important to network design and performance evaluation. Due to the distributed management of the Internet, an organization can only access its local information. To obtain the characteristics between networks require collaborations between organizations. However, commercial interests prohibit ISPs to exchange this type of information with their competitors. Network tomography tends to develop methods to obtain the characteristics by end-to-end measurements since those characteristics can help us to identify network problems and find solutions for future networks.

As similar techniques used in medical science, mine exploration, object detection, such as computed tomography and radar, network tomography relies on a trigger-response scheme to discover network characteristics. It sends probe packets (called probes later) from a node or a number of

nodes to an interested network with ongoing traffic, the probes head to a number of destined receivers, via the network. When probes arrive at the receivers, they carry the information about the network, which can only be extracted by specific statistical methods.

To determine the characteristics from observations, a probability model is normally selected to describe the corresponding characteristics of a link with some or all parameters undetermined. Network tomography in this circumstance investigates the methods and methodologies to identify those parameters from correlation observed from arrived probes. Two methods are used to send probes to receivers, i.e. multicast and unicast. The multicast approach is more scalable than the unicast one since it avoids the repeated transmission of the same packet on the same link. Due to this, we in this paper only consider the multicast situation, although it can be easily extended to the unicast situation.

A number of methods, including Expectation and Maximization algorithm (EM) and Bayesian estimation [1], have been proposed to determine those parameters. All those methods either use the iterative approximating approach to estimate the characteristics or are involved in solving a set

* Corresponding author. Tel.: +6126 2688171; fax: +6126 2688151.

E-mail addresses: weiping@cs.adfa.edu.au (W. Zhu), zgeng@math.pku.edu.cn (Z. Geng).

of high order polynomials [2,3]. No matter which method is used, the time spent on the estimation increases sharply as the size of the network being estimated, which may restrict the technique to be used in real-time situation. To overcome the problem, we need to search for other alternatives to speed up the process [3]. In this paper, we propose a bottom-up approach to estimate loss rates, which depends on the correlation observed between receivers to identify parameters. In contrast to other methods that aim to identify all parameters together, the proposed method takes a step-by-step approach to obtain the parameters one at a time from bottom-up. It starts from leaf links since the loss rate of a leaf link can be estimated directly from observations. Once the loss rates of leaf links are determined, the proposed method moves one level up along the multicast tree to estimate the loss rates of those links that connect to leaf links, the same principle is repeatedly applied at each level until it reaches the source. At each level, the proposed method relies on the observed differences between the receivers connected to a link and the receivers connected to the sibling brothers of the link to estimate the loss rate of the link. The proposed method only needs simple arithmetic calculation to determine loss rates, therefore, it is an analytical solution. As the method proposed in Ref. [4], the method proposed in this paper is a pseudo maximum likelihood estimator (PMLE), not an MLE, the results identified by this method is only a slightly smaller than that of an MLE in theory. In practice, the difference between them is negligible and it is consistent. Our simulation presented later in this paper shows the proposed method obtains identical results as the MLE.

The rest of the paper is organized as follows. In Section 2, we present the related work. We then introduce loss tomography and the principle used in statistical inference in Section 3. In Section 4, we detail the bottom-up approach presented in this paper with some examples. Section 5 presents the results of the inference algorithm based on the data collected from a simulation platform built on $ns-2$ [5]. Section 6 is devoted to concluding remark, it also contain our current and future work in line of measuring network performance.

2. Related work

Network tomography has a number of components for loss, delay, and bandwidth, respectively. Each component has its unique name to distinguish itself from others. Loss tomography, as named, aims to find loss rates of links. It depends on sending probes to the receivers attached to the end-nodes and apply the correlation observed by the receivers to identify the loss rates of those links that form a multicast tree [2,3,6–9]. Two methods are widely used to create correlated observations, i.e. multicast probes or unicast probes. A multicast-based method, as named, multicasts probes from a source to all receivers along a multicast

tree that covers the interested network on a specific basis, e.g. periodically or exponentially. The observations of the receivers that share the same parent or ancestor have strong correlation because of the intrinsic nature of multicast, which creates the foundation to determine the parameters of related links. On the other hand, the unicast-based approach targets those networks that do not support multicasting. To have correlated observations, the unicast approach uses various techniques to group a number of probes sent to different receivers together. For instance, one of the techniques called *packet-pair* has been used for loss tomography, that sends two packets, one after the other separated by a small amount of time, ϵ , from a source to two receivers that share a part of their paths from the source. If ϵ is small enough, there is a very low probability that a traffic surge could interrupt the packet pair. Thus, the two packets are expected to have similar experience on the shared paths. The difference observed by the receivers is most likely due to the different loss characteristics on the paths that are not shared by the two receivers. Then, similar inference techniques as those used in multicast-based methods are applied to identify loss rates on the shared path and not shared paths.

Statistical inference views each probe sent to receivers as a trial and what receivers observed from a trial as a sample. Since observations are only carried out at receivers, the samples collected from trials are incomplete with regards to the internal nodes. Statistical inference aims to uncover the loss rates of all links, including those that cannot be observed directly. So far, the methods proposed to discover the loss rates can be divided into two classes: classic statistics and Bayesian statistics, each class has its advantages and disadvantages in statistics.

Cáceres et al. are the pioneer to use the multicast-based approach to create correlation and subsequently find loss rates [2,10,11]. They assumed a Bernoulli loss model for a link, and derived a high order polynomial to describe the relation between a node and its children. By solving the polynomial, which normally requires an iterative procedure, the loss rates embedded in the polynomial can be identified. Both simulation and experiment studies on the Mbone show the feasibility and potential of this approach. The group also attempted to use multiple sources to cover a more general network [12]. This time, instead of the polynomial method, they used the EM and minimum variance weighted average (MVWA) to infer the loss rates. In the study, they assumed an i.i.d Bernoulli loss model for all links, and found all time the EM algorithm finds a solution, which is better than what MVWA did. Further, they found almost all time the EM converged at the global maximums, instead of local ones although that are possible as admitted [12].

Harfoush et al. [13] and Coates et al. [14] independently proposed the use of the unicast-based approach to discover link-level characteristics. Their simulations confirm the feasibility of this method. Coates and Nowak also suggested to use EM algorithm to estimate the correlation between

225 packet pairs for loss rates. Recently, Liang and Yu propose a
 226 PMLE to speed up the estimation process, in which they pair
 227 the observations of any two receivers together to form a
 228 paired likelihood, and then maximize all paired likelihood,
 229 instead of maximizing all possible joint probability [4].
 230 Although this method significantly reduce the time spent on
 231 estimation, it is still based on the iterative procedure to
 232 search for a solution in a multi-dimensional space: Guo and
 233 Wang [15] proposed a number of Markov Chain Monte
 234 Carlo (MCMC) algorithms to estimate loss and delay
 235 characteristics. Apart from loss characteristics, link delay is
 236 also attracted considerable attention. Shih and Hero [16]
 237 suggested a penalized maximum likelihood EM algorithm
 238 to identify the delay density functions. A common feature of
 239 those approaches is that they all use the iterative
 240 approximating approach to find a feasible solution, and
 241 the computation time increases exponentially as the number
 242 of hidden nodes/links, which makes those methods
 243 unscalable.

244 Apart from estimation methods, there are different views
 245 with regard to the model used to describe the characteristics
 246 of a link. Some researchers prefer to have complicated
 247 models that consider both temporal and spatial correlation
 248 since study shows some characteristics are temporal related
 249 [17]; while other researchers prefer to have simple ones. For
 250 instance, Coates et al. [3] believe Bernoulli model is good
 251 enough to describe link losses for network tomography, and
 252 they believe unless a high accuracy is absolutely necessary,
 253 one should try to use the model that is as simple as possible
 254 since simple assumptions about spatial and temporal
 255 independence often devise practical and scalable inference
 256 algorithm.

257 **3. Loss inference**

260 The multicast tree used to send probes to receivers can be
 261 abstracted by a three-element tuple (V, E, Θ) . The first two
 262 elements represent the nodes and links that have the same
 263 definitions as that in graph theory, i.e. $V = \{V_0, V_1, \dots, V_n\}$ is a
 264 set of nodes, which correspond to routers and switches in a
 265 network, $E = \{E_1, \dots, E_m\}$ is a set of links that connect the
 266 elements of V to form a network. While, $\Theta = \{\theta_1, \dots, \theta_m\}$ is
 267 an m -element vector, each for a link that is the parameters to
 268 be determined by statistical inference. For instance, if
 269 assuming the losses occurred on a link are independent, the
 270 Bernoulli model is adopted and Θ denotes the loss rates of
 271 links directly. If a Gaussian model is used, Θ denotes both
 272 means and variances.

273 When a probe is multicasted from the source to its
 274 receivers, the probe must first reach the root of the multicast
 275 tree before it is delivered to the receivers. Taking the extra
 276 leg into account, a multicast tree is a bit different from a
 277 regular tree at its root that has only a single child. However,
 278 as a regular tree, a multicast tree can be defined recursively,
 279 i.e. each sub-multicast tree has a root that has only one child
 280

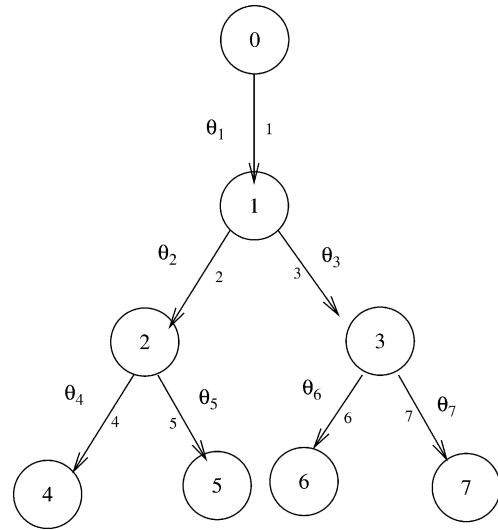


Fig. 1. Network structure.

281 that connects a normal tree. As a regular tree, we assign a
 282 unique number to each link, starting from 1, and a unique
 283 number to each node, starting from 0, the two sets of
 284 numbers map each other in the same way as a normal tree,
 285 e.g. link 1 connects node 1's parent (node 0) to node 1, link 2
 286 connects node 2's parent to node 2, and so on. Fig. 1 shows
 287 an example, apart from node 0 that is the root of the
 288 multicast tree, every node has one input link that has the
 289 same number as the node.

290 When a probe is sent to a multicast tree, each node in the
 291 multicast tree has only two possible outcomes: observed or
 292 missed. Let 1 denote the observed outcome, and 0 denote
 293 the other. In addition, let X denote a node and its state, F_x
 294 denotes the parent of X (if X has). The model used to
 295 describe the loss rates of the link connecting F_x to X is a
 296 conditional probability, $P(X|F_x)$. Obviously, we have
 297 $P(X=0|F_x=0) = 1$. However, what we are interested is

$$P(X = 0|F_x = 1)$$

298 the loss rate of link X . Statistical inference is used here to
 299 estimate the loss rate from observations, in particular for
 300 those links that cannot be observed directly. Each
 301 observation collected by receivers corresponds to a set of
 302 joint probabilities that lead to the observation. For instance,
 303 with Bernoulli loss model and a multicast tree as shown in
 304 Fig. 2, when an observation $r = (X_2=0, X_3=1)$ is collected,
 305 we have the joint probability as follows:

$$P(r; \Theta) = (1 - \theta_1)\theta_2(1 - \theta_3).$$

306 If an observation $r = (X_2=0, X_3=0)$ is obtained, the joint
 307 probability turns to the following form:

$$P(r; \Theta) = \theta_1 + (1 - \theta_1)\theta_2\theta_3$$

308 where the first term on the right hand side (RHS) represents
 309 the loss occurred on link 1 since $P(X_2=0|X_1=0) = 1$ and
 310 $P(X_3=0|X_1=0) = 1$, the second term of the RHS represents
 311 the probe passed link 1, but lost by links 2 and 3

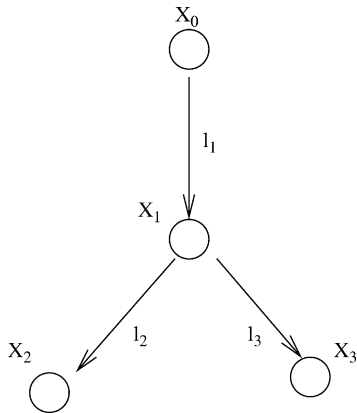


Fig. 2. A simple multicast tree.

simultaneously that leads $X_2=0$ and $X_3=0$. The latter example shows an observation can have a number of joint probabilities that can all lead to the same observation. In general, given the loss model and the multicast tree structure, one is able to construct joint probabilities from observations as shown in the above example. With a large number of observations, statistical inference aims to identify the unknown parameters embedded in the joint probabilities. In this circumstance, loss tomography actually is a parametric estimation with incomplete data in statistics. If the maximum likelihood estimate (MLE) is applied to determine the parameters, it can be written in a log-likelihood format

$$\arg \max_{\Theta} L(\Theta) = \arg \max_{\Theta} \sum_{r \in \Omega_R} n(r) \log P(r; \Theta) \quad (1)$$

where $n(r)$ is a function that counts the number of occurrences of observation r in the trials. Formula (1) shows the problem in fact is an optimization problem, and there are a number of methods, e.g. neural net, Monte-Carlo, EM, that can be used to find a solution for (1). If there are n parameters to be identified, the above methods use iterative approximating procedures to search for the maximum point in the n -dimensional space until it is converged. As previously stated, the search can take a long time to converge, and may converge at a local maximum, instead of the global one.

4. Bottom-up approach

What we are concerned here is whether there are other alternatives to conduct the estimation, which are simple, efficient and accurate, in particular if we want to use it for network controls

If assuming the losses occurred on two serially connected links are independent, i.e. spatial independent, and identical distributed (i.i.d), when the multicast approach is used to create correlation among receivers, we can have a simple and efficient approach to estimate the loss rates of a network

that considers the unique nature of this problem and takes a bottom-up approach to conduct its estimation. For the bottom approach, the loss rates of all leaf links can be estimated directly from observations. Once, the loss rates of all leaf links have been identified, the proposed method moves one level up to estimate the loss rates of the links that are parents of the leaf links. In this level, each link plus the subtree connected to the link is regarded as a virtual link, then the loss rate of the virtual link can be estimated as a leaf link. By knowing the loss rates of leaf links, we are able to obtain the loss rate of the parent link from the loss rate of the virtual link. This process is continued until it reaches the source. The following three subsections are used to detail the proposed algorithm for leaf links, internal links and top link, respectively.

4.1. Leaf link

The proposed method starts to estimate the loss rates of those leaf links that have all their sibling brothers' observations available. Let X be the link connecting node X , and let B_x denote the sibling brothers of link X , which is a binary set recording the observations of those receivers attached to those brothers. Each element in the set represents the observation of a receiver for a probe, 1 means the receiver observed the probe, 0 means otherwise. Let S_{B_x} represent the observation of the sibling brothers of X , which is defined as

$$S_{B_x} = \begin{cases} 1, & \exists i, i \in B_x, i = 1 \\ 0, & \forall i, i \in B_x, i = 0 \end{cases} \quad (2)$$

Formula (2) shows if at least one of the elements in B_x is 1, $S_{B_x} = 1$, that also implies the parent of X observed the probe. Since S_{B_x} is independent from X , the loss rate of link X can be derived:

$$\begin{aligned} P(X = 0 | F_x = 1) &= P(X = 0 | F_x = 1, S_{B_x} = 1) \\ &= P(X = 0 | S_{B_x} = 1) \end{aligned} \quad (3)$$

Recall that $n(y)$ is a count function that records the number of y appeared in the trials. We can use $n(\cdot)$ to estimate $P(X = 0 | S_{B_x} = 1)$

$$P(X = 0 | S_{B_x} = 1) = \frac{\sum_{B_x} n(X = 0, S_{B_x} = 1)}{\sum_{B_x} n(S_{B_x} = 1)} \quad (4)$$

Note that $n(S_{B_x} = 1) = n(X = 0, S_{B_x} = 1) + n(X = 1, S_{B_x} = 1)$. For example, to estimate the loss rate of link 4 of Fig. 1, we have

$$\begin{aligned} P(X_4 = 0 | X_2 = 1) \\ &= \frac{n(X_4 = 0, X_5 = 1)}{n(X_4 = 0, X_5 = 1) + n(X_4 = 1, X_5 = 1)} \end{aligned}$$

which is identical to the formula derived by Cáceres et al. from a high order polynomial [2,10].

When the loss rates of all leaf links are obtained by the above method, the dimensions of the solution space are halved. We then can either use the traditional EM algorithm to search for the parameters of the other links or move one level up as we propose in the follows.

4.2. Internal link

For an internal link, X , once the loss rates of all its children, which can be a set of leaf links, or a set of subtrees, or a combination of the previous two, have been estimated, the loss rate of the subtree rooted at node X can be estimated by summing of the products of the loss rates of those links that form the *cuts* of the subtree. A cut of a subtree is a group of links that can separate the subtree horizontally into two parts. For instance, for the subtree rooted at node 1 of Fig. 1, there are four cuts, which are

1. link 2 and link 3;
2. link 3, link 4 and link 5;
3. link 2, link 6 and link 7; and
4. link 4, link 5, link 6 and link 7;

since each of them can horizontally cut the subtree into two pieces.

To identify all cuts of a multi-level subtree is a tedious task. However, given the cuts of all subtrees connected to a tree, one is able to identify all the cuts of the tree, which are the combination of those links that connect the tree to its subtrees and the cuts identified from those subtrees. In the above example, there are four cuts, the first one consists of the links that connect to the two subtrees rooted at node 2 and 3, respectively; for the other three cuts, one consists of link 3 and the cut of the subtree rooted at node 2, one consists of link 2 and the cut of the subtree rooted at node 3, the last consists of the cuts of the subtrees rooted at node 2 and 3. The purpose of identifying the cuts of a subtree is to obtain the loss rate of the subtree. To avoid repeatedly calculating the loss rates of the same subtrees, a general formula is developed that takes into account of the recursive nature of cuts. Let C_x denote the set of links that connect node X to its children, let $f_i(0)$ denote the loss rate of link i and let $f_i(1)$ be the loss rate of the subtree rooted at node i . Then, the loss rate of the subtree rooted at node X is equal to the sum of the follows:

- the product of the loss rates of those links that connect X to its children, $\prod_{i \in C_x} f_i(0)$;
- the sum of the products obtained from the combination of the loss rates of some links connecting X to its subtrees, denoted by SC_x , and the loss rates of those subtrees in $C_x \setminus SC_x$, where SC_x can be empty.

If there are n subtrees connected to node X , there are $1 + \sum_{i=1}^n C_n^i = 2^n$ terms in the formula. Let g_x represent

the loss rate of a subtree rooted at node X , then we have

$$g_x = \sum_{i_1=0}^1 \cdots \sum_{i_n=0}^1 f_{s_1}(i_1) f_{s_2}(i_2) \cdots f_{s_n}(i_n) \quad (5)$$

where s_1 to s_n denote the n subtrees connected to X . $f_x(\cdot)$ is determined by the following rules:

- For a leaf link, X , after its loss rate is estimated, we set $f_x(0) = \hat{P}(X=0|P_{a_x}=1)$ and $f_x(1)=0$ since a leaf link does not connect to any subtree.
- For a non-leaf link, Y , after estimating $P(Y=0|F_Y=1)$, we set $f_y(0) = P(Y=0|F_Y=1)$ that is the loss probability of link y and $f_y(1) = g_y[1 - f_y(0)]$ that is the product of the pass rate of link Y and the loss rate of the subtree rooted at node Y .

When the loss rates of all links on this level have been estimated, the proposed method moves one level up to estimate the loss rates of those links in that level. The process is continued until all links have been estimated. For example, the loss rate of the subtree rooted at node 1 of Fig. 1 can be estimated by

$$g_1 = f_2(0)f_3(0) + f_2(1)f_3(0) + f_2(0)f_3(1) + f_2(1)f_3(1). \quad (6)$$

The four terms on the RHS correspond to the four cuts previously listed.

The bottom-up approach ensures when it is estimating the loss rate of an internal link, X , the loss rates of all the links in the subtree rooted at X are available. Then, link X plus the subtree rooted at it can be regarded as a virtual link, denoted as V_x , and the loss rate of the virtual link can be estimated from observations. From the viewpoint of F_x , the parent of X , V_x is strongly related to B_{V_x} , the sibling brothers of V_x . The view of V_x for a probe sent to it is defined in a similar manner as Eq. (2)

$$S_{V_x} = \begin{cases} 1, & \exists i, i \in R(X), i = 1 \\ 0, & \forall i, i \in R(X), i = 0 \end{cases} \quad (7)$$

where $R(X)$ denotes those receivers attached to the subtree rooted at node X . Applying (4) here, we have

$$\hat{P}(V_x = 0|F_x = 1) = \frac{\sum_{B_{V_x}} n(S_{V_x} = 0, S_{B_{V_x}} \neq 0)}{\sum_{B_{V_x}} n(S_{B_{V_x}} \neq 0)} \quad (8)$$

Since

$$P(V_x = 0|F_x = 1) = g_x + (1 - g_x)P(X = 0|F_x = 1)$$

where g_x is the loss rate of the subtree rooted at node X . Then, we have

$$\hat{P}(X = 0|F_x = 1) = \frac{1}{1 - g_x} [P(V_x = 0|F_x = 1) - g_x] \quad (9)$$

If $1 - g_x = 0$, there is no definition for the above formula because the loss rate of the subtree is 1. That means the receivers attached to the subtree have not observed a single

probe. To avoid this, before estimating the loss rates of a network after a series of trials, we need to have a pre-processing to eliminate those links whose loss rates cannot be estimated from the observations. We first delete all the leaf

$$P(X_2 = 0|X_1 = 1) = \frac{P(V_2 = 0|X_1 = 1) - g_2}{1 - g_2} = \frac{P(X_4 = 0 \wedge X_5 = 0|X_6 = 1 \vee X_7 = 1) - P(X_4 = 0|X_2 = 1)P(X_5 = 0|X_2 = 1)}{1 - P(X_4 = 0|X_2 = 1)P(X_5 = 0|X_2 = 1)} \quad (12)$$

links that have not received any probes since their loss rates are inestimable. In other words, the loss rate of such a link can be regarded as one. If all leaf links connected to the same parent node are deleted, the link connecting the grand-parent to the parent node is also deleted since there is no observation that can assist us to estimate the loss rate of the link. In addition, if an internal node X has only one child C , we delete the node X and connect the parent of X , F_x , to C directly since $P(C=0|X=1)$ and $P(X=0|F_x=1)$ are not identifiable. The pre-processing aims to eliminate those subtrees, including leaf links that have not observed any probes sent to them. The loss rates of those links or subtrees can be regarded as one, but cannot be estimated by Eq. (9). This also ensures the above formulae are properly defined. For example, if receiver four of Fig. 1 misses all probes, the loss rate of link 5, $\theta_5 = P(X_5 = 0|X_2 = 1) = P(X_5 = 0|X_4 = 1)$ is undefined. In the pre-processing, we replace the subtree rooted at node 2 by a new node, called X_{25} , and its observation is equal to the observation received by X_5 . Then, $P(X_{25}=0|X_1=1)$ can be estimated, but remember $P(X_{25}=0|X_1=1) \neq P(X_2=0|X_1=1)$ and $P(X_{25}=0|X_1=1) \neq P(X_5=0|X_2=1)$. The RHSs are inestimable in this situation.

4.3. An example

We use Fig. 1 as an example to demonstrate how the proposed method identifies the loss rates of the links embedded in the network. Since the multicast tree has three levels, the proposed method takes three steps to complete the task.

4.3.1. Leaf links

There are four receivers attached to the multicast tree, the loss rates of the four leaf links, θ_4 , θ_5 , θ_6 , and θ_7 , can be obtained directly from observation by Eq. (3), which are

$$\begin{aligned} \theta_4 &= P(X_4 = 0|X_2 = 1) = P(X_4 = 0|X_5 = 1) \\ \theta_5 &= P(X_5 = 0|X_2 = 1) = P(X_5 = 0|X_4 = 1) \end{aligned} \quad (10)$$

$$\theta_6 = P(X_6 = 0|X_3 = 1) = P(X_6 = 0|X_7 = 1)$$

$$\theta_7 = P(X_7 = 0|X_3 = 1) = P(X_7 = 0|X_6 = 1) \quad (11)$$

The above estimations can be carried out in parallel. Before moving to the second level, $f_4(\cdot)$, $f_5(\cdot)$, $f_6(\cdot)$, and $f_7(\cdot)$ are set accordingly.

4.3.2. Second level

Once the loss rates of all leaf links have been obtained, we can estimate the loss rates of links 2 and 3. Using Eq. (9), we have the estimate of the loss rate of link 2, which is

Since the terms of RHS are either known or observable, the LHS is estimable. Similarly, θ_3 can be estimated by the following

$$\begin{aligned} \theta_3 &= P(X_3 = 0|X_1 = 1) \\ &= \frac{P(X_6 = 0, X_7 = 0|X_4 \vee X_5 = 1) - g_3}{1 - g_3} \end{aligned} \quad (13)$$

where $g_3 = f_6(0)f_7(0) = P(X_6 = 0|X_3 = 1)P(X_7 = 0|X_3 = 1)$. To prove (13), we use Dawid's notation [18].

Proof. Since $(X_4, X_5) \perp\!\!\!\perp (X_6, X_7)$, i.e. given X_1 , the observation of X_4 and X_5 is independent to that of X_6 and X_7 , and $(X_4 \vee X_5 = 1)$ implies $X_1 = 1$, we have

$$\begin{aligned} P(X_6 = 0, X_7 = 0|X_1 = 1) &= P(X_6 = 0, X_7 = 0|X_1 = 1, (X_4 \vee X_5 = 1)) \\ &= P(X_6 = 0, X_7 = 0|X_4 \vee X_5 = 1) \end{aligned} \quad (14)$$

In addition

$$\begin{aligned} P(X_6 = 0, X_7 = 0|X_1 = 1) &= P(X_3 = 0|X_1 = 1) \\ &+ P(X_3 = 1|X_1 = 1)P(X_6 = 0|X_3 = 1)P(X_7 = 0|X_3 = 1) \end{aligned} \quad (15)$$

Using (10) and (11) to replace $P(X_6=0|X_3=1)$ and $P(X_7=0|X_3=1)$ from the above two equations, we obtain (13). \square

After setting $f_2(\cdot)$ and $f_3(\cdot)$, we move to the top level.

4.3.3. Top level

$$\theta_1 = P(X_1 = 0|X_0 = 1) = \frac{P(V_1 = 0|X_0 = 1) - g_1}{1 - g_1} \quad (16)$$

where

$$g_1 = f_2(0)f_3(0) + f_2(1)f_3(0) + f_2(0)f_3(1) + f_2(1)f_3(1) \quad (17)$$

Note that $P(X_0=1)=1$, thus, we can estimate θ_1

Proof. Since

$$\begin{aligned} P(V_1 = 0|X_0 = 1) &= P(X_1 = 0|X_0 = 1) + P(X_1 = 1|X_0 = 1)g_1 \\ &= P(X_1 = 0|X_0 = 1) + (1 - P(X_1 = 0|X_0 = 1))g_1 \end{aligned} \quad (18)$$

solving the equation, we have (16). \square

4.4. Discussion

There are two issues that need to be clarified for the proposed method: one is whether the virtual link concept used above can be applied to a number of serial connected links, instead of a subtree; the other is whether the method is an MLE, if not, what is the difference between these two.

To illustrate the concept used in the bottom-up method, two diagrams are presented in Fig. 3, in which the triangle shape connected to node X denotes the subtree rooted at the node. Given the observations of B_x and the loss rate of the subtree rooted at node X , the loss rate of link X can be estimated by using formula (13), in which link X plus the subtree is regarded as a virtual link, as shown in Fig. 3(a), called subtree approach. Alternatively, a path connecting X to a receiver, consisting of a number serially connected links, can be regarded as a virtual link, as shown in Fig. 3(b), called serial link approach. Then, based on the correlation between the newly defined virtual link and the observation of B_x , we can also estimate the loss rate of link X . For instance, to estimate the loss rate of link 2 of Fig. 1 links 2 and 4 can be considered as a virtual link, then the loss rate of the virtual link is equal to

$$P(X_4 = 0|X_1 = 1) = P(X_2 = 0|X_1 = 1) + (1 - P(X_2 = 0|X_1 = 1))P(X_4 = 0|X_2 = 1)$$

Considering the correlation between the virtual link and B_x , i.e. X_6 and X_7 , we have

$$P(X_4 = 0|X_1 = 1) = P(X_4 = 0|X_6 = 1 \vee X_7 = 1)$$

Solving the above two equations, we have

$$\theta_2 = P(X_2 = 0|X_1 = 1) = \frac{P(X_4 = 0|X_6 = 1 \vee X_7 = 1) - P(X_4 = 0|X_5 = 1)}{1 - P(X_4 = 0|X_5 = 1)}$$

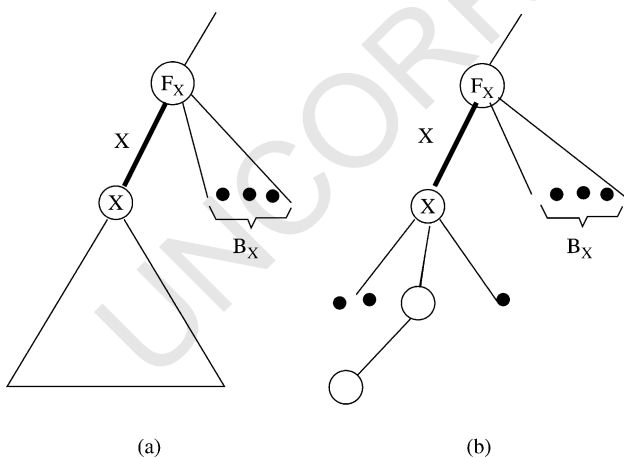


Fig. 3. Two different approaches.

Similarly, we have

$$\theta_3 = P(X_3 = 0|X_1 = 1) = \frac{P(X_6 = 0|X_4 = 1 \vee X_5 = 1) - P(X_6 = 0|X_7 = 1)}{1 - P(X_6 = 0|X_7 = 1)}$$

where

$$P(X_4 = 0|X_6 = 1 \vee X_7 = 1) = \frac{n(X_4 = 0 \wedge (X_6 = 1 \vee X_7 = 1))}{n(X_6 = 1 \vee X_7 = 1)}$$

The terms of the RHS can be determined by observations, where the items occurred in $n(\cdot)$, such as X_4 in the numerator, have the required values; while the others can take any possible values. For instance, $n(X_6 = 1, X_7 = 1)$ for the multicast tree shown in Fig. 1 sums all four type of observations up, i.e. $\langle X_4 = 0, X_5 = 0, X_6 = 1, X_7 = 1 \rangle$, $\langle X_4 = 0, X_5 = 1, X_6 = 1, X_7 = 1 \rangle$, $\langle X_4 = 1, X_5 = 0, X_6 = 1, X_7 = 1 \rangle$ and $\langle X_4 = 1, X_5 = 1, X_6 = 1, X_7 = 1 \rangle$. Therefore, we have θ_2 and θ_3 . Comparing the two approaches, the serial link approach is simpler, but it is less accurate than the subtree approach since the pass rate of subtree X is higher than that of any path connecting X to a receiver.

The proposed method is a PMLE, not an MLE, because when it estimates the loss rate of link X , it ignores the impact of those probes that arrive at node F_x , but do not receive by any receivers attached to the subtree rooted at F_x , including those receivers attached to the subtree rooted at node X and those in B_x . This type of losses can be divided into two groups: one is for those probes that pass link X but lost by the subtree attached to it; the other for the rest, i.e. those are lost at link X . The latter losses should be considered in the estimation. However, the proposed method simply ignores them because it cannot distinguish the latter from the former. If still using Fig. 1 as an example, when we estimate the loss rate of link 2, θ_2 , the proposed method does not consider the impact of those probes that arrive at node 1, but lost by the subtree rooted at node 1. If adding this part to the first term of the numerator of Eq. (9), it will become an MLE. However, for an observation that can be created either by a simultaneous loss involved by a number of links or by a loss of the parent link of the previous links, the latter's probability is much higher than the former. That explains why in our simulation study, the proposed method achieves identical results as MLE. The advantage of the bottom-up method lies on its efficiency and simplicity. The proposed method can be easily extended to the highly demanded distributed scheme, in which the receivers are grouped according to the multicast tree and perform the inference from bottom-up, i.e. those groups that do not share the same parent can carry out their inference independently. Once all groups that have the same parent complete their inferences, they form a new group to estimate the loss rate of the parent link.

5. Simulation study

To demonstrate its effectiveness of the proposed method, we conducted two round tests on a simulation environment built on ns2. The first round has eight nodes connected by seven links, named 1–7, into a tree structure, as shown in Fig. 1. In the first round, link 1 had 3 Mbps of bandwidth, 2 ms of propagation delay; links 2 and 3 also had 3 Mbps of bandwidth, but 10 ms of propagation delay; the other four links had 1.5 Mbps of bandwidth and 10 ms propagation delay. All nodes have a FIFO queue and except node 1 has a queue with a limit of 20 packets, all other nodes can at most queue 10 packets at a time. The droptail policy is employed by all nodes to handle congestion, i.e. when a queue is full, newly arrived packets were dropped. Probe packets, 40 bytes each, were periodically multicasted from the root to the receivers attached to the leaf nodes. The background traffic consists of:

1. two TCP streams with window size=50 and packet size=1 KB flow from node 0 to nodes 4 and 5, respectively;
2. four exponential distributed on–off UDP streams;
 - one burst stream with burst period=400 ms, idle period=300 ms, bit_rate=1000 k, and packet size=200 B flows from node 0 to 4;
 - one burst stream with burst period=300 ms, idle period=300 ms, bit_rate=800 k, and packet size=200 B flows from node 0 to 5;
 - one burst stream with burst period=300 ms, idle period=200 ms, bit_rate=400 k, and packet size=500 B flows from node 1 to 6;
 - one burst stream with burst period=200 ms, idle period=200 ms, bit_rate=400 k, and packet size=500 B flows from node 1 to 7.
3. one FTP stream flows from node 0 to 4 with window size=60 and packet size=600; and
4. three FTP streams flow from node 0 to nodes 5, 6 and 7, respectively, with window size=60, and packet size=800;

where the burst periods and idle periods yield exponential distribution, and the numbers provided above are the means of the corresponding exponential distribution. Except 2. that started at 0 s and suspended at 50 s, and resumed at 80 s, all other streams started flow from 0 to 95 s.

What we were interested in this study is to find out the packets loss rate at each link by end-to-end measurement. $\theta_i, i \in \{1, \dots, 7\}$ in Fig. 1 represents the loss rate of link i . A multicast agent is added on the root node (0) to multicast probe packets on a regular basis to the four leaf nodes. A sequence number is attached to each probe packet, then based on the sequence number, a receiver can identify whether a probe packets is lost, if so, its position in the probe stream. During the experiments, we conducted an inference every 5 s based on the data collected at the four

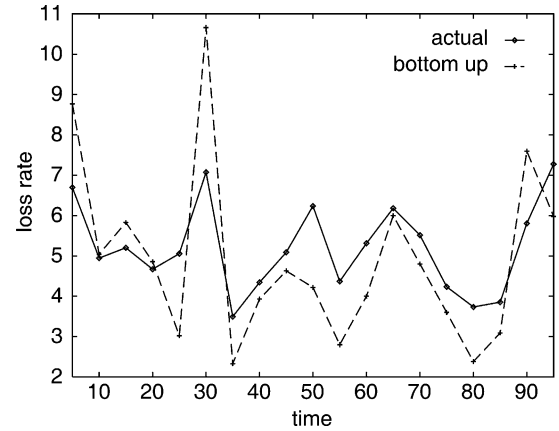


Fig. 4. Loss rate on link 1 with probe interval=0.02 s.

receivers, the simulator uses the same interval to collect the actual link-level data, packet sent and dropped, at every node. We call the data collected by the simulator actual result.

In the experiment, the inferred results on links 2, 3, 6 and 7 match the true results perfectly since these links were lightly loaded. Figs. 4 and 5 shows the difference between inferred result and true result on link 1 and 4, respectively. Although there are some differences between the inferred and actual results, the inferred results correctly show the loss trend of the background traffic, in particular, when stream 2 was suspended.

There are 16 nodes in the second round, the nodes are connected into a binary tree as shown in Fig. 6, the speed and delay used for a link is attached to the link. Unlike to the first round, each node has a queue that can hold 2000 packets, and random early detection (RED) is used by each node to control congestion. Apart from this, we set the loss rate for each link in this round that assumes a hierarchical network structure: the top links (links 1–3) are located in the core network that has the lowest loss rate 1%; the links in the next level (links 4–7) are located in regional networks whose loss rates

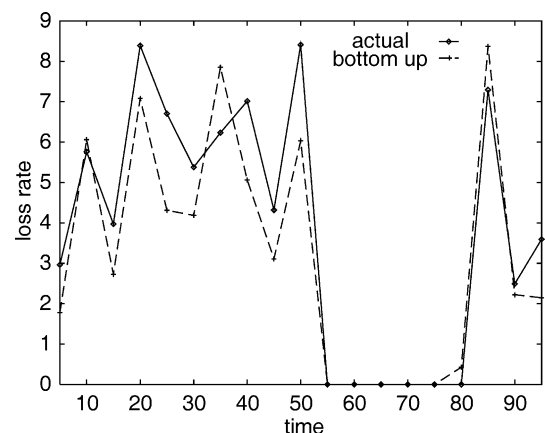


Fig. 5. Loss rate on link 4 with probe interval=0.02 s.

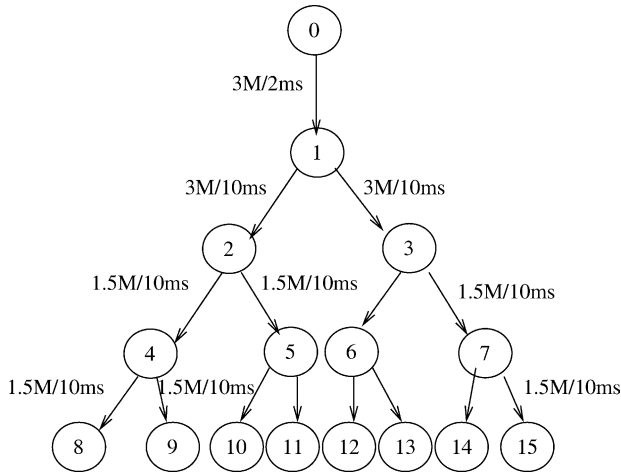


Fig. 6. Network structure used in second round.

are set to 2%; links 8–15 are located in local networks with a loss rate of 3%. This setting provides a more flexible environment to test the proposed method under various conditions. In the first round, we can only test and verify the method with losses created by congestion. In the second, the two type of losses, one is created by congestion (run out of buffer), the other is created by corruption, can be tested. The background traffic added in this round is more complex than the previous one, which includes:

1. TCP sources that continuously send packets of size 1024 bytes, with a window size=50, are located at:
 - node 0 that forwards packets to node 8;
 - node 0 that forwards packets to node 9;
 - node 2 that forwards packets to node 10;
 - node 2 that forwards packets to node 11.
2. A TCP source at node 2 sends packets of size 600 B with window size=60 to node 9.
3. A burst stream with burst period=400 ms, idle period=300 ms, bit_rate=1000 k, and packet size=200 B flows from node 0 to 8, which is stopped from 50 to 80 s
4. A burst stream with burst period=400 ms, idle period=200 ms, bit_rate=1000 k, and packet size=200 B flows from node 0 to 9.
5. A burst stream with burst period=300 ms, idle period=200 ms, bit_rate=400 k, and packet size=500 B flows from node 1 to 10.
6. A burst stream with burst period=200 ms, idle period=200 ms, bit_rate=400 k, and packet size=500 B flows from node 1 to 11;
7. A burst stream with burst period=300 ms, idle period=200 ms, bit_rate=400 k, and packet size=500 B flows from node 3 to 15.
8. A burst stream with burst period=300 ms, idle period=200 ms, bit_rate=800 k, and packet size=500 B flows from node 3 to 14.

9. A burst stream with burst period=300 ms, idle period=100 ms, bit_rate=600 k, and packet size=500 B flows from node 3 to 13.
10. A burst stream with burst period=300 ms, idle period=200 ms, bit_rate=400 k, and packet size=500 B flows from node 3 to 12.
11. A burst stream with burst period=400 ms, idle period=300 ms, bit_rate=1000 k, and packet size=200 B flows from node 4 to 8.
12. A burst stream with burst period=500 ms, idle period=300 ms, bit_rate=1000 k, and packet size=200 B flows from node 4 to 9.

The load added at the tree structure is unbalanced; the left subtree has a heavy load and the right has a light one. For an overloaded network, its losses are dominated by congestion; while, the losses occurred in a light loaded network are more likely due to corruption. Our purpose here is to study the robustness of the proposed method in balanced and unbalanced situations.

Apart from the background traffic, a multicast agent is added at node 0, which periodically sends probes to receivers. Two frequencies are used to send probes to receivers in this round to test the consistency of the proposed method: one sends 50 probes/s, the other sends 100 probes/s. A consistent method should reduce its variation with the increase of samples. Therefore, the result obtained from the higher frequency one should be better than the other. We present the result of this round in Figs. 7–10. There are 30 sub-figures consisting of 15 pairs in these figures, one for a link. We pair the same link with different probing frequencies together and put them side by side, in which the left one is for 50 probes/s and the right one is for 100 probes/s. The pairs start from link 1 and end at link 15 in a sequential order. Each figure shows the actual loss occurred on a link against the estimated loss for the link. For light loaded links, i.e. links 3, 6, 7, and 12–15, the estimates fit to the actual loss and varies around the losses we set up in the round. For the left subtree, the estimates made by the proposed method fit to the actual loss very well, which show the proposed method can adapt to various conditions. Since different flows exist in this round test, the probe flow based on UDP may not suffer the same loss rate as those TCP flows. Therefore, the estimate error may be due to either the estimate error or probe loss rate differs from the loss rate of the background traffic. For instance, there are obvious difference between the estimate loss rate and the actual one for links 1 and 2 (Fig. 7), despite the difference is reduced as the increase of probing frequency. The flows of the background traffic added on these two links are dominated by TCP flows that have congestion control capability, while the probe flow does not have the similar capability. That partially explains why the estimate loss has a quick up and down feature while the actual loss are relative smooth.

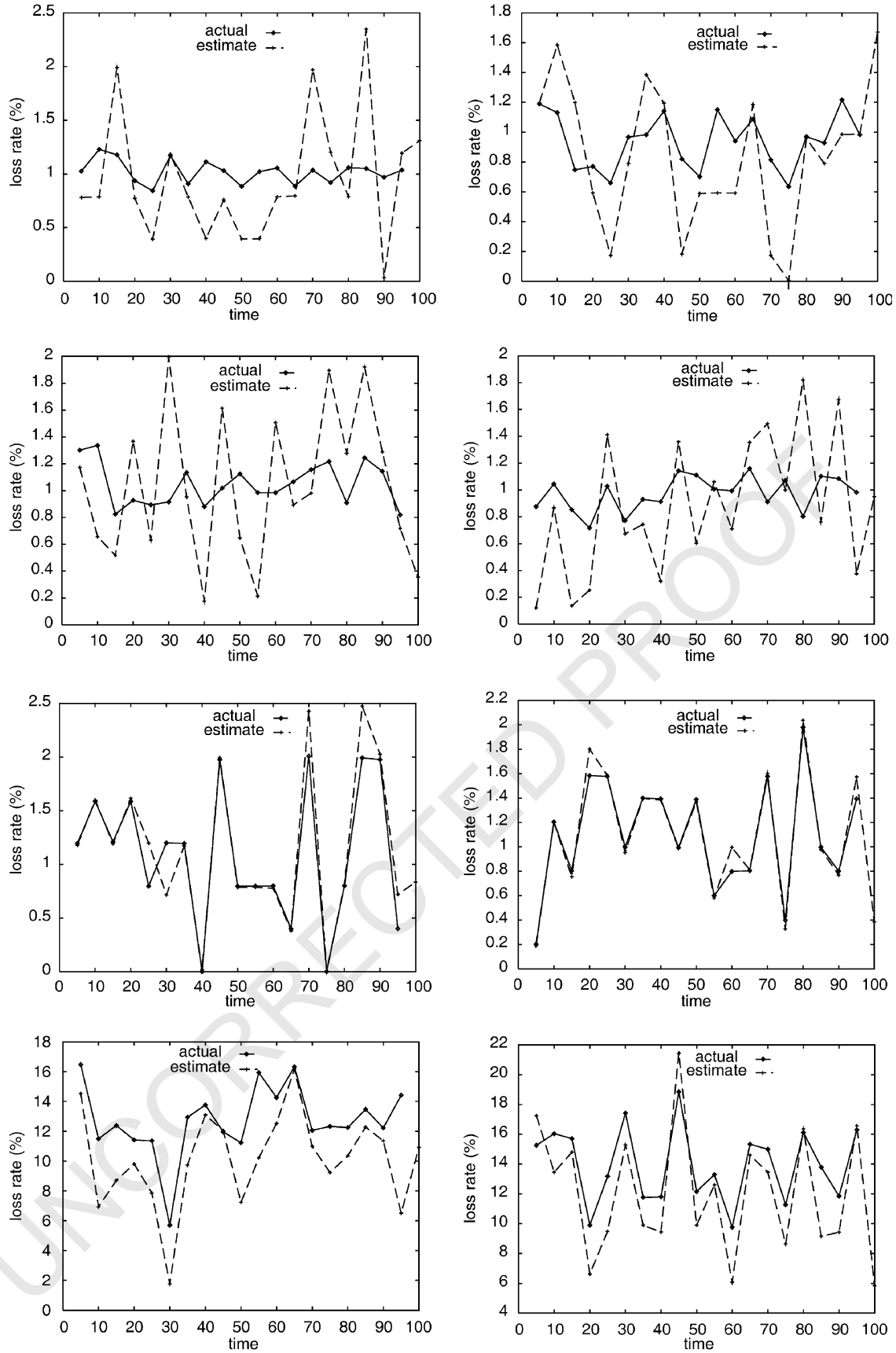


Fig. 7. Actual loss vs. estimate loss, links 1–4.

1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064

1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120

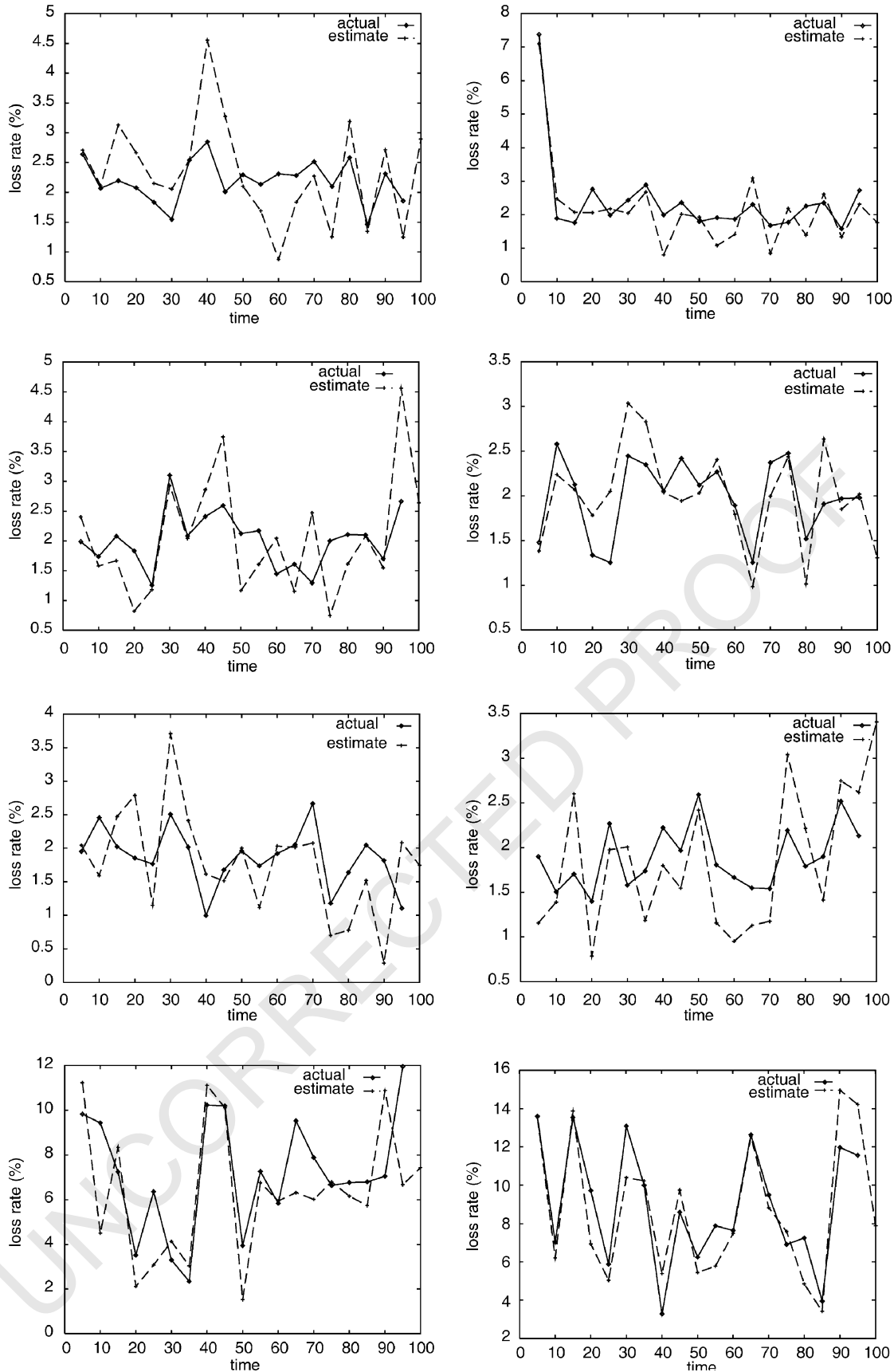


Fig. 8. Actual loss vs. estimate loss, links 5–8.

1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176

1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232

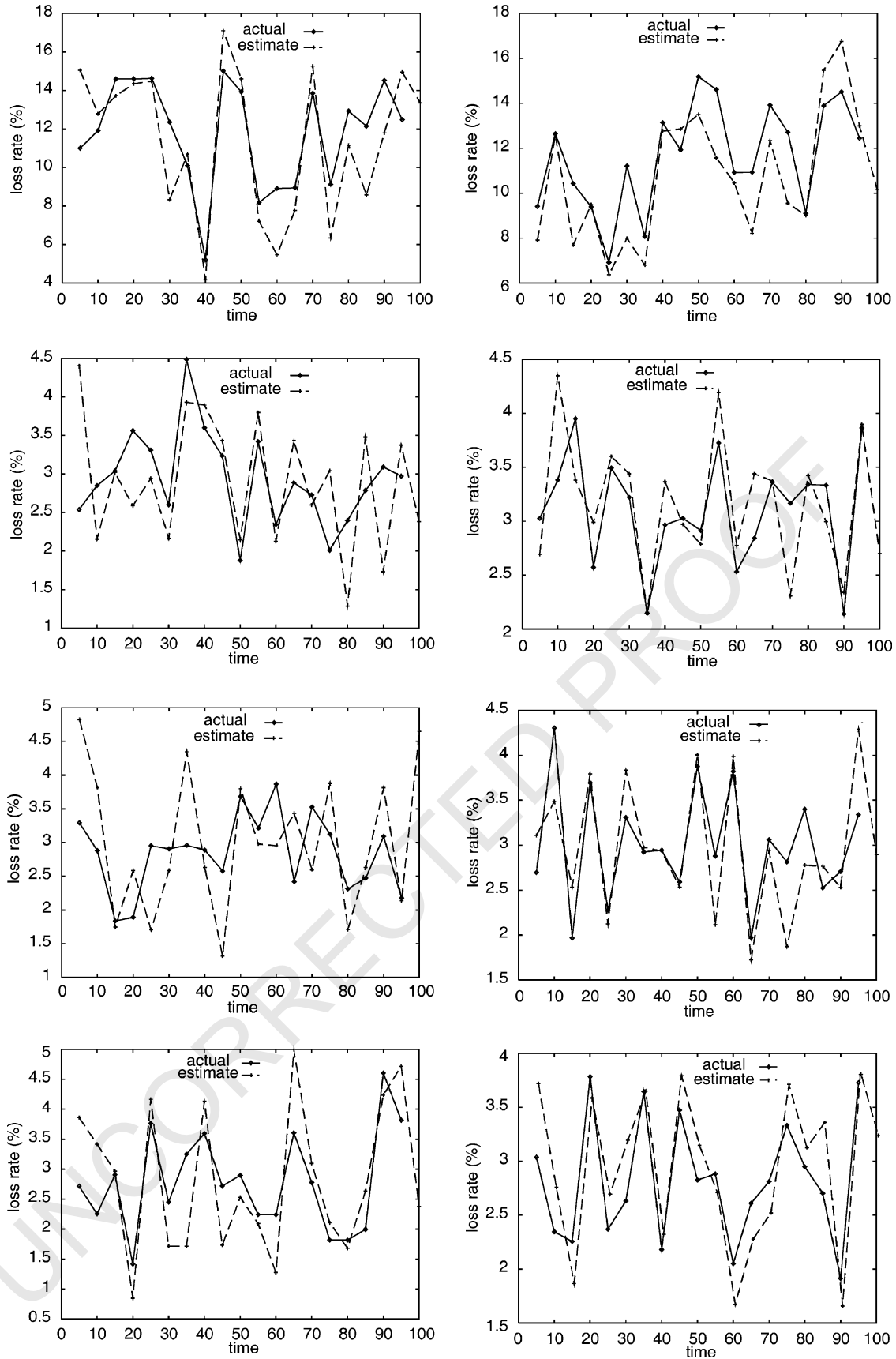


Fig. 9. Actual loss vs. estimate loss, links 9–12.

1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288

1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344

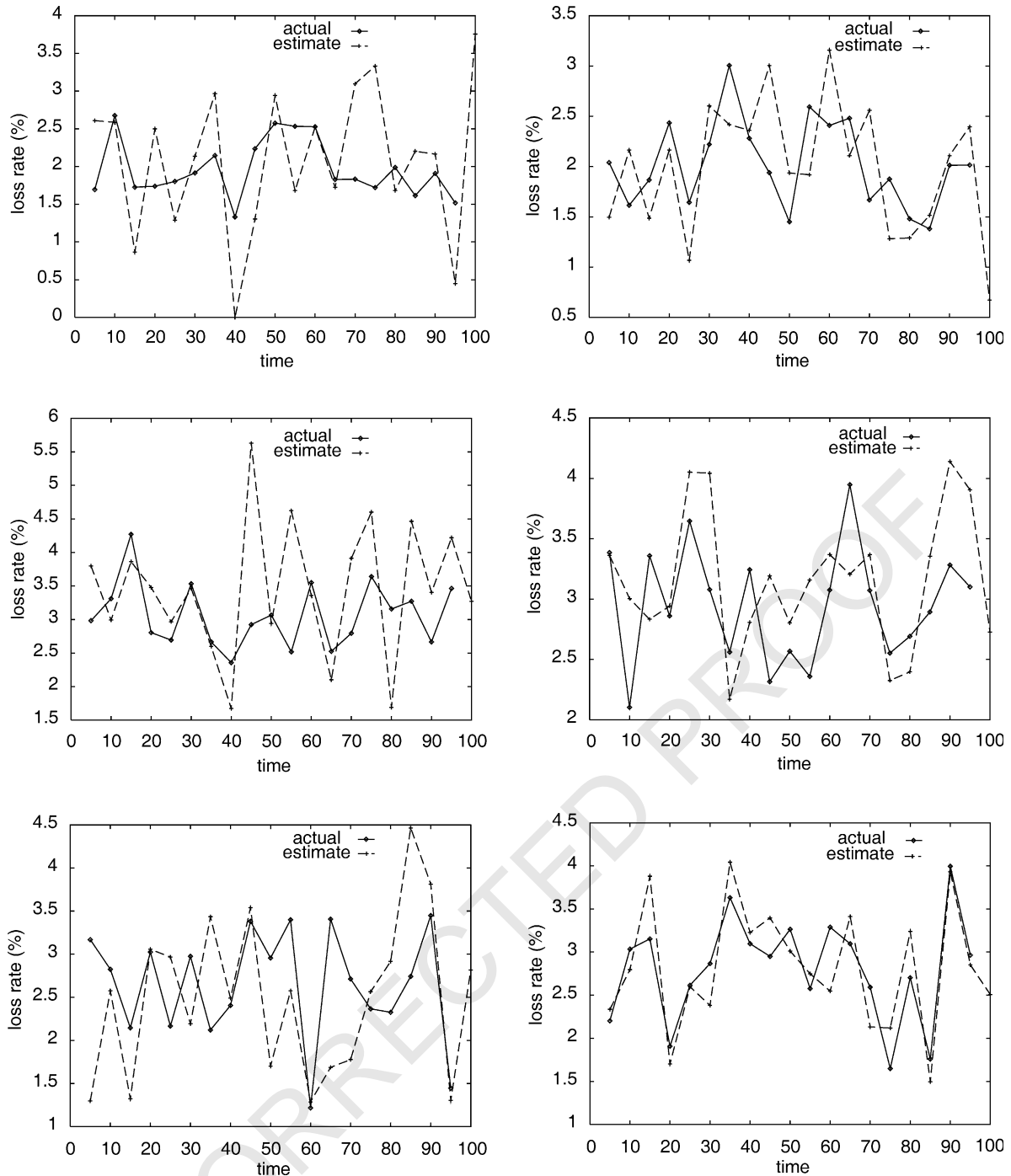
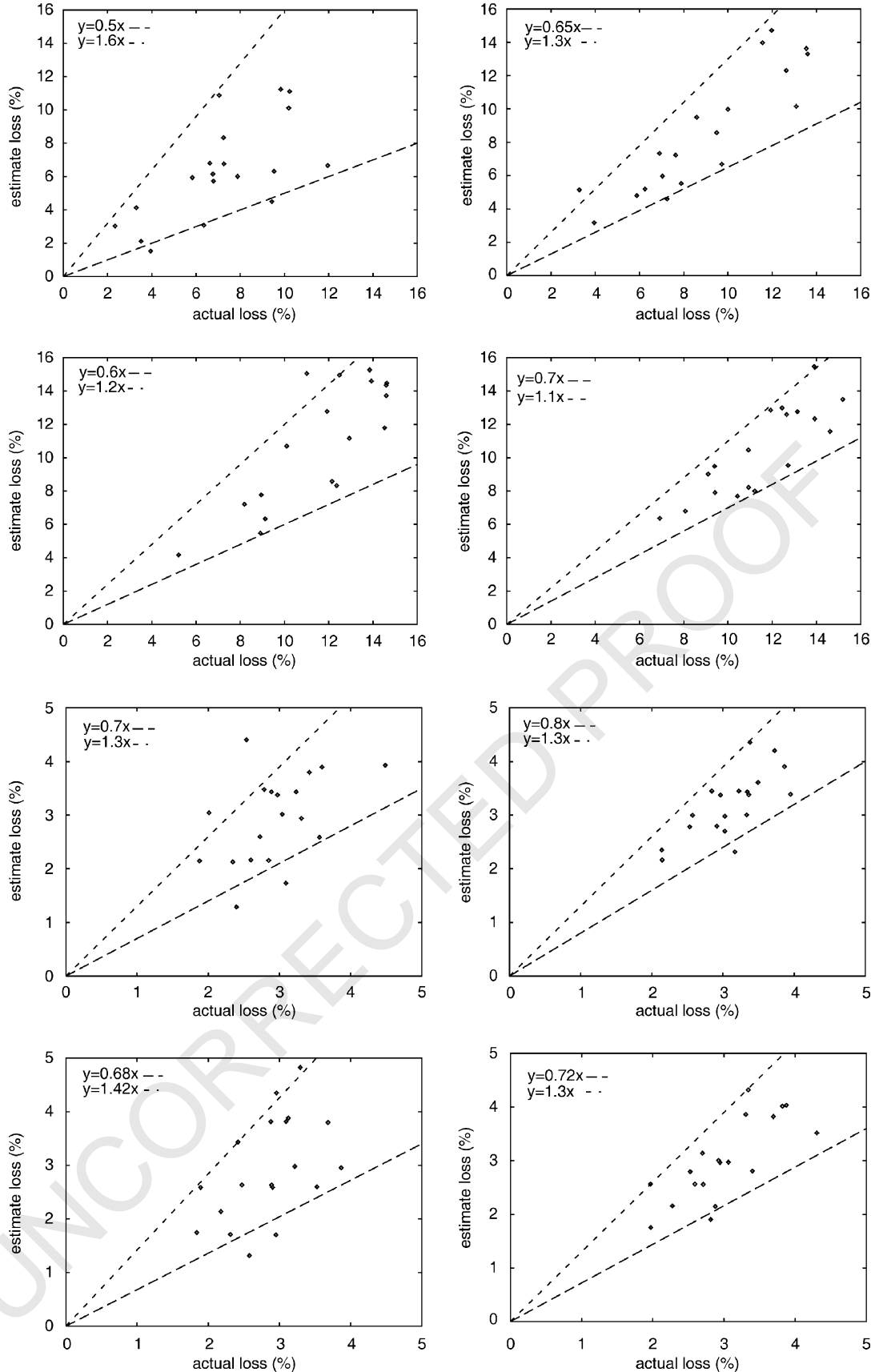


Fig. 10. Actual loss vs. estimate loss, links 13–15.

To show the consistency of the method, we present four scatter plots of the actual loss vs. estimated loss for links 8–11 in Fig. 11. As previous figures, we pair the scatter plots according to links and the left one is for the probing frequency of 50 probes/s. There are two lines in each plot to show the accuracy of the estimate results. Ideally, the estimate points should fall on the line connecting the top right to bottom left. Practically,

a good estimate should have its estimate points closely scattered along the line. In comparison between two plots, a more accurate estimate should have a smaller angle between the two lines than the less accurate one. The plots on the right hand column obviously are more focused than those on the left hand, which confirm that the result obtained by using high frequency has less variance than the other. In other words, using

1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512



1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568

Fig. 11. Scatter plots for links 8–11.

the proposed method, the estimate result approaches to actual loss with the increase of samples.

6. Conclusion

Network tomography depends on statistical inference to identify network characteristics which cannot be observed directly. MLE is one of the most popular strategy used in inference. For a large network with hundred of links, using iterative approximation to infer the parameters of the links can take considerable amount of time since the iterative approximation needs to search for a feasible solution in a complex solution space. Apart from that, the solution identified by a method may not be the global maximum since it may trap into a local optimum. To overcome the first problem, we in this paper present a simple bottom-up approach to estimate the loss rates of a network that in principle applies the observed correlation between a link and its sibling brother to identify the loss rate of the link. More importantly, rather than identifying all parameters together, the proposed method separates this process into a number of steps, depending on the levels of the multicast tree used to send probes and at each step it estimates the parameters for one link only. The proposed method starts from the bottom to determine the loss rates of leaf links. It then moves one level up to repeat the same operation until reaches the root. The advantage of this approach relies on its simplicity, efficiency and consistency. In fact, the proposed approach is an analytical solution. Comparing with the MLE in simulations set up on $ns-2$, we find the proposed method achieves identical results as the MLE. The form or shape of the multicast tree used to send probes to receivers is another issue that requires further study, which is related to the network topology and identifiability that determines the number of receivers and their locations.

References

- [1] W. Zhu, Using Bayesian networks on network tomography, *Computer Communications* 26 (2) (2003). 1625
- [2] R. Cáceres, N.G. Duffield, J. Horowitz, D. Towsley, Multicast-based inference of network-internal loss characteristics, *IEEE Transactions on Information Theory* 1999; 45. 1626
- [3] M. Coates, A. Hero, R. Nowak, B. Yu, Internet tomography, *IEEE Signal Processing Magazine* 19 (3) (2002). 1627
- [4] G. Liang, B. Yu, Maximum pseudo likelihood estimation in network tomography, *IEEE Transactions on Signal Processing* 51 (8) (2003). 1628
- [5] The network simulator 2, Technical Report, www.isi.edu/nsnam/ns2. 1629
- [6] Felix: Independent monitoring for network survivability, Technical Report, [ftp://ftp.bellcore.com/pub/mwg/felix/index.html](http://ftp.bellcore.com/pub/mwg/felix/index.html). 1630
- [7] Ipma: Internet performance measurement and analysis, Technical Report, <http://www.merit.edu/ipma>. 1631
- [8] J. Mahdavi, V. Paxson, A. Adams, M. Mathis, Creating a scalable architecture for internet measurement, *INET'98*. 1632
- [9] Surveyor, Technical Report, <http://io.advanced.org/surveyor>. 1633
- [10] R. Cáceres, N.G. Duffield, S.B. Moon, D. Towsley, Inference of internal loss rates in the MBone, *IEEE/ISOC Global Internet'99* 1999. 1634
- [11] R. Cáceres, N.G. Duffield, S.B. Moon, D. Towsley, Inferring link-level performance from end-to-end multicast measurements, Technical Report, University of Massachusetts, 1999. 1635
- [12] T. Bu, N. Duffield, F.L. Presti, D. Towsley, Network tomography on general topologies, *SIGCOMM* 2002. 1636
- [13] K. Harfoush, A. Bestavros, J. Byers, Robust identification of shared losses using end-to-end unicast probes, Technical Report BUCS-2000-013, Boston University, 2000. 1637
- [14] M. Coates, R. Nowak, Unicast network tomography using EM algorithms, Technical Report TR-0004, Rice University, September 2000. 1638
- [15] D. Guo, X. Wang, Bayesian inference of network loss and delay characteristics with applications to tcp performance prediction, *IEEE Transactions on Signal Processing* 51 (8) (2003). 1639
- [16] M.-F. Shih, A.O. Hero, Unicast-based inference of network link delay distribution with finite mixture models, *IEEE Transactions on Signal Processing* 51 (8) (2003). 1640
- [17] M. Yajnik, S. Moon, J. Kurose, D. Towsley, Measurement and modelling of the temporal dependence in packet loss, *IEEE Infocom* 1999. 1641
- [18] A.P. Dawid, Conditional independence in statistical theory, *Journal of the Royal Statistical Society Series A* (41) (1979). 1642