Hindawi
Mathematical Problems in Engineering
Volume 2017, Article ID 6342421, 17 pages
https://doi.org/10.1155/2017/6342421

Research Article

Congested Link Inference Algorithms in Dynamic Routing IP Network

Yu Chen[1,2] and Zhe-min Duan[1]

[1]*Northwestern Polytechnical University, Xi'an 710072, China*
[2]*Zhengzhou University of Aeronautics, Zhengzhou 450015, China*

Correspondence should be addressed to Yu Chen; chenyu@zua.edu.cn

Received 13 August 2016; Accepted 28 November 2016; Published 18 January 2017

Academic Editor: Yaguo Lei

The performance descending of current congested link inference algorithms is obviously in dynamic routing IP network, such as the most classical algorithm CLINK. To overcome this problem, based on the assumptions of Markov property and time homogeneity, we build a kind of Variable Structure Discrete Dynamic Bayesian (VSDDB) network simplified model of dynamic routing IP network. Under the simplified VSDDB model, based on the Bayesian Maximum A Posteriori (BMAP) and Rest Bayesian Network Model (RBNM), we proposed an Improved CLINK (ICLINK) algorithm. Considering the concurrent phenomenon of multiple link congestion usually happens, we also proposed algorithm CLILRS (Congested Link Inference algorithm based on Lagrangian Relaxation Subgradient) to infer the set of congested links. We validated our results by the experiments of analogy, simulation, and actual Internet.

1. Introduction

Transmission performance of links in IP network is very important to network users and operators. With the rapid expansion of IP network scales [1], the method of artificial periodic inspection has not been suitable to current IP network needed. The active measurement methods of congested link inference in IP network are based on the ICMP (Internet Control Message Protocol), which does not involve personal privacy, and only adopt a fewer E2E multiple-path measurements (snapshots), E2E path properties, and topology information of IP network can be obtained. With the help of Bayesian theory, link properties can be inferred in real-time. Comparing with the passive measurement methods based on the SNMP (Simple Network Management Protocol), the active measurement methods have many advantages. But, the congested link inference algorithms based on active measurements also have some disadvantages. (1) Parts of inference algorithms assume that links traversed by E2E paths do not change during the process of congested link inference, such as the most classical algorithm CLINK [2], which infers the set of congested links by static Bayesian Network Model (BNM). In addition, Qiao et al. [3] introduced a transition probability to improve the inference performance based on the Dynamic Bayesian Network Model (DBNM) in varying parameter. Because most of IP network autonomous regions (AS) adopt the dynamic routing algorithm (such as OSPF), when bandwidths of links are limited, routing of E2E paths will change [4, 5]. When inferring the set of congested links based on BNM, changes of routing will bring the changes of BNM's structures, not only the parameters. (2) In order to reduce algorithm complexity, some inference algorithms assume the number of congested links does not exceed certain rates [6]. But, with the expansion of IP network scale, multiple link congested concurrent phenomena usually appear. With the increasing number of congested links, when congested links reach a certain number, the inference performance of current algorithms will obviously descend because of the minimum set cover method, such that CLINK algorithm will only infer a most likely congested shared link along a congested path. (3) In addition, current congested link inference algorithms did not deeply research performance impacts under the different link coverage. Due to the tomography technology, using less E2E path measurement to infer link properties as much as

possible, the inference performance will be affected when the link coverage is not enough.

Aiming at disadvantages of current inference algorithms in dynamic routing IP network, a kind of Variable Structure Discrete Dynamic Bayesian (VSDDB) network model was established, and, according to its simplified model based on assumptions of Markov property and time homogeneity, we proposed a kind of Improved CLINK (ICLINK) algorithm and a kind of Congested Link Inference algorithm based on Lagrangian Relaxation Subgradient (CLILRS). Experiments verified the inference performance in different link congested scenarios.

In this paper, we briefly summarized our contributions listed as follows:

(1) Aiming at the dynamic routing IP network, VSDDB network model was established. Based on the assumptions of Markov property and time homogeneity, VSDDB network model was simplified.

(2) Based on the simplified model, the congested link inference algorithms of ICLINK and CLILRS were proposed, which efficiently solve congested link inference problem in dynamic routing IP network.

(3) Under the different link congested scenarios, experiments of analogy, simulation, and actual Internet verified the inference performance of algorithms proposed in this paper.

The rest of this paper is organized as follows: Section 2 summarized works related to our researches. In Section 3 we introduced the inference problems in dynamic routing IP network and built VSDDB simplified model. In Section 4 we described the design methods of algorithm in detail. In Section 5, we evaluated the performance of congested link inference algorithm. Finally, Section 6 concluded our paper.

2. Related Work

Since 1996, Vardi [7] first put forward network tomography technology similar to medical tomography in the IP network link properties inference. The inference of internal link properties from End-to-End (E2E) measurements is called network tomography. It requires solving a system of equations relating the E2E measurements with the link properties either in linear or in Boolean algebra.

By using multicast probing packet [8–10] or multicast-like (multiple clusters of unicast) [11–13], path properties can be measured. When inferring link loss rate by each E2E path loss rate, it needs to be put into complex hardware infrastructure investment [14] and requires high clock synchronization to measure each E2E path property. [15] Due to the security reasons, most routers support unicast more than multicast [16]. Insufficiency of E2E measurement information will cause the coefficient matrix to underdetermine system linear equations, which is unable to accurately calculate the link loss rate unique solutions. With the expansions of IP network scales, the inverse problem of linear equations will become more complex and the real-time performance of algorithm can not be guaranteed, even leading to failure. Duffield [13] gives a different congested state threshold according to loss rate of each E2E path. According to the differences of loss rates, Gu et al. [17] proposed a kind of optimal detection scheme according to unicast delay based on Fischer information matrix theory.

The second kind of methods divides the properties of paths or links into binary variables and obtains Boolean algebraic values of link status to locate the congested links. Nguyen and Thiran [2] built Boolean algebra linear equations between status of paths and corresponding traversing links. Padmanabhan et al. [18] and Duffield [13] simplified the congested link inference methods by using Boolean algebra model. Padmanabhan et al. [18] proposed a kind of MCMC algorithm without using prior probabilities. Duffield [13] proposed a kind of SCFS algorithm based on consistent prior probabilities in small value p_0. In actual IP network, link congestion probabilities in the backbone are significantly smaller than the access network, and, when $p_0 = 0.2$, the inference DR (Detection Rate) of SCFS algorithm is only 30%. The algorithm accuracy of CLINK [2] has a greater degree improvement than MCMC and SCFS. Especially, when there are many congested links in IP network, DR of CLINK is obviously higher.

Although Nguyen and Thiran thought that routing of E2E path will change when inferring the set of congested links, they also regarded that congestion link will continue for several hours and did not consider the performance impacts for the link coverage during the process of congested link inference. When the link coverage appears changing, the inference performance will descend. Aiming at the routing changing during the process of learning congested link prior probabilities, Qiao et al. [3] treated the routing changing as noise interference, by adding transition probabilities in static BNM, which improved the inference precision. However, because of the dynamic routing algorithm in IP network AS (es), link coverage will change and performance of current inference algorithms will descend based on the static BNM.

In this paper, we built a kind of Variable Structure Discrete Dynamic Bayesian (VSDDB) network model and proposed improved algorithms to identify congested links based on the simplified VSDDB network model in dynamic routing IP network. Experiments of analog, simulation, and actual network, respectively, verified the inference performance of algorithms proposed in this paper.

3. VSDDB Network Inference Model

The IP network link congestion usually happens, and the routing of each E2E path will dynamically change due to some factors, such as the available bandwidth changing of links. In order to accurately locate the most likely congested links in IP network, firstly, we build a directed acyclic graph model $G = (\nu, \varepsilon)$ with the help of graph theory, where the node set of ν can be denoted as the routers, switches, hosts, and so forth. And the directed edge set of ε can be denoted as the connected links of nodes. In order to build congested link inference model, we make Definition 1 to express the status of E2E paths and corresponding traversing links.

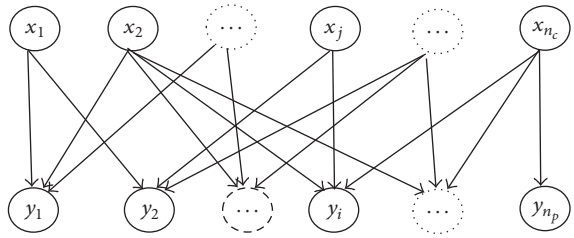

Figure 1: BNM of IP network.

Definition 1. The variables representing the status of the E2E paths are $\mathbf{Y} = (y_1, \ldots, y_i, \ldots, y_{n_p})$, and the variables representing the status of the links are $\mathbf{X} = (x_1, \ldots, x_j, \ldots, x_{n_c})$. If P_i is congestion, $y_i = 1$; and $y_i = 0$ otherwise. In a similar way, if the link l_j is congestion, $x_j = 1$; and $x_j = 0$ otherwise.

Where n_p is the total number of E2E paths, n_c is the total number of all traversing links. In BNM of IP network, the evidence nodes are the state variables of E2E path properties, the hidden nodes are the state variables of link properties, and the directed edges represent the relationships among the E2E paths and corresponding traversing links. BNM of IP network is described as in Figure 1.

During inferring congested links, if some E2E path is "good," each link traversed by this path is "good." According to this rule, during the process of learning link prior probabilities and inferring the set of congested links, we do not need considering those "good" paths and corresponding traversing links. We can remove the evidence node y_i and its corresponding hidden nodes x_j, as well as the directed edges connecting them of BNM. So, we make Definition 2 as follows.

Definition 2. BNM of IP network will be converted into Rest BNM (RBNM) after removing evidence nodes of $y_i = 0$ (good E2E paths), hidden nodes (links traversed by good E2E paths), and corresponding directional edges.

When inferring the set of congested links under BNM, according to the dynamic routing algorithm strategy in IP network, for N times of E2E path snapshots, links traversed by E2E paths will be changed due to some factors, such as the physical link cut or link bandwidth limited. Therefore, during N times of E2E path snapshots, dynamic change of routing matrix can happen. So, we introduced Definition 3.

Definition 3. If the structure (including parameter) of a discrete static BNM will change in different time, this kind of model is called Variable Structure Discrete Dynamic Bayesian (VSDDB) network model [19].

Therefore, in dynamic routing IP network, during inferring the set of congested links, the routing changing usually happens; in order to simplify the solution process, we introduce two fundamental assumptions to simplify VSDDB model.

Assumption 4 (first-order Markov property). Once current system status is given, status in the past and future, respectively, is independent.

Assumption 5 (time homogeneity). Conditional probabilities of nodes in each time slice and transition probabilities among each time slice do not change.

To construct the routing matrix D of IP network, we introduced Definition 6 as follows.

Definition 6. Each row of IP network routing matrix D is E2E path P_i ($i = 1, 2, \ldots, n_p$); each column is all links l_j ($j = 1, 2, \ldots, n_c$) arranged according to every hop count from start of router to end. When some link l_j is contained in some E2E path P_i, corresponding element value $d_{ij} = 1$; otherwise, $d_{ij} = 0$.

In the process of learning congestion prior probabilities of links in IP network, we can obtain t number static BNMs if the routing matrix D changes t times. That is, to N times of snapshots, we regard a certain number of snapshots with the same routing status as a piece of time slice. According to Assumptions 4 and 5, we can simplify DBNM into two static BNMs, respectively, which are the initial time slice T^1 during the prior probability learning process and the time slice T^2 of the present inference moment; two time slices are connected by state variables of common links, respectively, existing in time slices T^1 and T^2.

Based on the VSDDB simplified model, we proposed an improved algorithm of identifying congested links in dynamic routing IP network. During the process of inferring congested links, first, we carry out N times of E2E path properties snapshots, to learn link congested prior probabilities according to E2E path properties measurements and corresponding routing matrix D in time slices T^1 and T^2. Then, based on the simplified VSDDB network model, according to $N + 1^{\text{th}}$ time E2E path snapshots results in the present inference moment, we infer the set of congested links. E2E paths and corresponding traversing links in the present inference moment can be obtained from the congested BNM in the time slice T^2 after removing nodes and directed edges representing good paths and corresponding traversing links; the RBNM and corresponding rest congested matrix of IP network in the present inference moment can be obtained. VSDDB network model is listed as in Figure 2.

In Figure 2, assuming links of $l_h, l_j, \ldots$, and l_k exist in time slices T^1 and T^2, these links will make up the node interfaces between the two time slices T^1 and T^2. The state variables of congested path are $y_i^{T^t} = 1$ (evidence nodes), and the state variables of links $x_j^{T^t}|_{x_j^{T^t}=\{0,1\}}$ (hidden nodes) in each piece of time slice build the corresponding static BNM. Each static BNM in the two time slices constitutes VSDDB network model through the node interfaces.

4. Congested Link Inference Algorithm

4.1. Constructing the Congested Routing Matrix. After obtaining links traversed by E2E paths in large scale IP network, we can construct the routing matrix D^{T^t} of IP network in every time slice T^t, $t = \{1, 2\}$. In order to reduce the

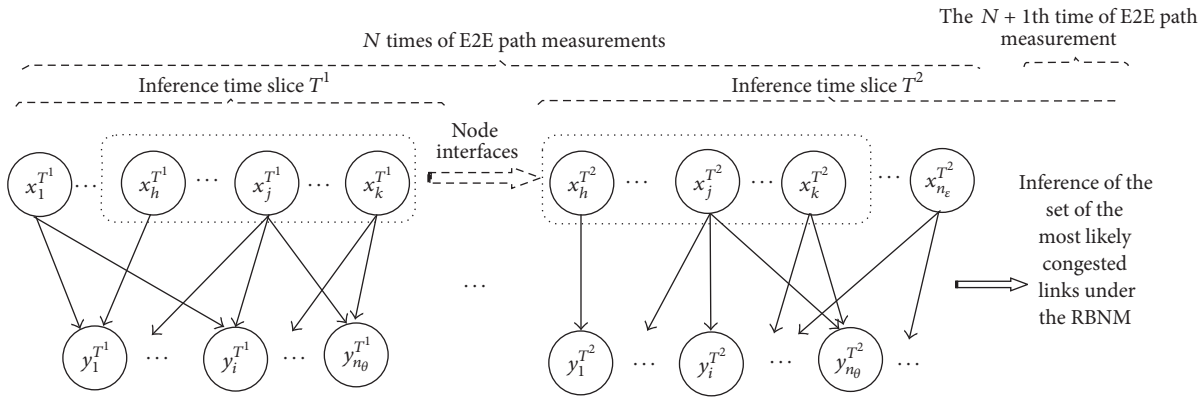

Figure 2: Simplified VSDDB network model. Note: in N times of snapshots, the number of E2E congested paths is n_θ, the number of traversing links in time slices T^1 and T^2, respectively, is $n_\varepsilon^{T^1}$ and $n_\varepsilon^{T^2}$, the number of common links (node interfaces) in both time slices T^1 and T^2 is n'_ε, and the number of congested paths and corresponding traversing links, respectively, is n'_θ and n''_ε in the present inference moment.

number of active measurements, we can make the primary row transformation to D^{T^t} and remove linear dependence path row to get the uncorrelated matrix D'^{T^t}. After removing linear dependence path rows, characters and column number of D^{T^t} will not change; that is, the number of links that inference algorithm covered does not change.

When solving the link congested prior probabilities through N times of E2E path snapshots, if the congestion times of E2E path do not exceed *path-congested-threshold*, we regard that this path is good during the process of learning congested link prior probabilities. In this paper, default setting is *path-congested-threshold* = 0; that is, if some E2E path shows one time congestion during N times of snapshots, this path is congested, at least showing one congested link traversed by this path. Otherwise, if some path is always good in N time of snapshots, the status of path is good, and its corresponding traversing links are all good, the corresponding row and column can be removed from uncorrelated routing matrix D'^{T^t}. After removing, we can get the rest of congested routing matrix $D_d^{T^t}$ in time slice T^t and obtain the uncorrelated congested matrix $D_d'^{T^t}$.

4.2. Solving the Congested Link Prior Probabilities. In time slice T^t ($t = \{1, 2\}$), relationships between the transmission rate of E2E congested path and corresponding traversing links meet the linear equations (1) [20].

$$\lg \Psi_i^{T^t} = D_{d_{ij}}'^{T^t} \lg \sum_{j=1}^{n_\varepsilon^{T^t}} \varphi_j^{T^t}, \tag{1}$$

where $\Psi_i^{T^t}$ is the transmission rate of i^{th} path, $\varphi_j^{T^t}$ is the transmission rate of j^{th} link, and $D_{d_{ij}}'^{T^t}$ is the element value of the decorrelation congested matrix. $n_\varepsilon^{T^t}$ is the number of congested links. According to Definition 1, when j^{th} link is traversed by i^{th} path P_i, $D_{d_{ij}}'^{T^t} = 1$; $D_{d_{ij}}'^{T^t} = 0$ otherwise. In early congested link inference algorithms, researchers translate the loss rate of each E2E path into transmission rate; after taking logarithm to (1), the linear equations of solving link transmission rate can be constructed. But, it is hard to ensure the real-time because the process of solving equations needs inverse operation; in large scale IP network, it can lead to failure. Therefore, we can adopt Boolean algebra to express status of E2E paths and links. Boolean algebra equations between E2E paths and corresponding traversing links are listed as in the following:

$$y_{i'} = \bigvee_{j=1}^{n_\varepsilon^{T^t}} x_j^{T^t} \cdot D_{d_{ij}}'^{T^t}, \tag{2}$$

where the symbol of "$\vee$" is binary maximum operator and $y_{i'}$ is the variable value of some linearly independent E2E path. Equations (2) meet descriptions between E2E paths and corresponding traversing links. In order to calculate congested link prior probabilities, we can take mathematical expectation to (2) and obtain the following:

$$\begin{aligned} E\left[y_{i'}\right] &= E\left[\bigvee_{j=1}^{n_\varepsilon^{T^t}} x_j^{T^t} \cdot D_{d_{ij}}'^{T^t}\right] = P\left[\bigvee_{j=1}^{n_\varepsilon^{T^t}} x_j^{T^t} \cdot \left(D_{d_{ij}}'^{T^t} = 1\right)\right] \\ &= 1 - P\left[\bigvee_{j=1}^{n_\varepsilon^{T^t}} x_j^{T^t} \cdot \left(D_{d_{ij}}'^{T^t} = 0\right)\right] \\ &= 1 - \prod_{j=1}^{n_\varepsilon^{T^t}} \left(1 - p_j^{T^t}\right)^{D_{d_{ij}}'^{T^t}}, \end{aligned} \tag{3}$$

where $n_\varepsilon^{T^t}$ is the number of all links traversed by congested paths and $E[y_{i'}]$ is the congestion probability $P_{i'}$ of E2E paths in $D_d'^{T^t}$. We can get its value through taking average of n times

of snapshots performance measurements and express it with $\overline{y}_{i'}$. After taking logarithm to (3), we can get the following:

$$\lg\left(1-\overline{y}_{i'}\right)=\sum_{j=1}^{n_\varepsilon^{T^t}} D_{d_{ij}}^{\prime T^t}\cdot\left[\lg\left(1-p_j^{T^t}\right)\right]. \tag{4}$$

In tomography methods, we usually adopt small quantities of E2E path measurements to cover links as much as possible. And after twice getting rid of correlation, the number of E2E paths corresponding to $D_{d_{ij}}^{\prime T^t}$ in (4) may be far less than the number of links. Therefore, the matrix coefficient of (4) may be underdetermined. In order to solve unique solution of (4), we need to fill up full rank operation to the coefficient matrix. The full rank coefficient matrix is shown as in (5). Because each row of coefficient matrix $D_{d_{ij}}^{\prime T^t}$ expresses E2E congested paths, we can regard two congested paths as one congested path through the binary max operation for corresponding traversing links in two paths; to construct getting rid of correlation expansion paths of $D_{d_{ij}}^{\prime T^t}$, we can express each expansion path as $D_{d_{kj}}^{\prime T^t}$. After combining with $D_{d_{ij}}^{\prime T^t}$ and $D_{d_{kj}}^{\prime T^t}$ and getting rid of correlation again, we can get the full rank coefficient matrix $\mathrm{D}_{d_{jj}}^{\prime\prime T^t}$ with the rank of n_ε; the size of k depends on the value $n_\varepsilon^{T^t}-n_{\theta'}^{T^t}$; $n_{\theta'}^{T^t}$ is the number of congested paths in matrix $D_{d_{ij}}^{\prime T^t}$ in time slice T^t.

$$D_{d_{jj}}^{\prime\prime T^t}=\begin{pmatrix} D_{d_{ij}}^{\prime T^t} \\ D_{d_{kj}}^{\prime T^t}\end{pmatrix}_{n_\varepsilon\times n_\varepsilon}. \tag{5}$$

But not all expansion paths can make the submatrix $\mathrm{D}_{d_{kj}}^{\prime T^t}$ in full rank. So, we also need to carry out getting rid of correlation until selecting suitable expansion paths to get full rank matrix $D_{d_{jj}}^{\prime\prime T^t}$. Equations of solving the congested link probabilities are listed as follows:

$$\lg\left(1-\overline{y}_{il}\right)=\sum_{j=1}^{n_\varepsilon^{T^t}} D_{d_{kj}}^{\prime T^t}\cdot\left[\lg\left(1-p_j^{T^t}\right)\right], \tag{6}$$

where $\overline{y}_{il}$ can be obtained by the binary max operation to path state variables y_i and y_l. And $k=j-i$, after filling up the full rank; vector expression equation is given as follows:

$$\begin{aligned}
&\begin{bmatrix}\mathbf{y}\\ \mathbf{y}'\end{bmatrix}=\mathrm{D}_{d_{jj}}^{\prime\prime T^t}\left[\lg\left(1-\mathbf{p}\right)\right]^T\\
&\mathbf{y}=\left[\lg\left(1-\overline{y}_1\right),\lg\left(1-\overline{y}_2\right),\ldots,\lg\left(1-\overline{y}_{n_{\theta'}^{T^t}}\right)\right]^T,\\
&\mathbf{y}'=\left[\lg\left(1-\overline{y}_{12}\right),\ldots,\lg\left(1-\overline{y}_{1n_{\theta'}^{T^t}}\right),\lg\left(1-\overline{y}_{23}\right),\ldots,\right.\\
&\quad\left.\lg\left(1-\overline{y}_{2n_{\theta'}^{T^t}}\right),\ldots,\lg\left(1-\overline{y}_{(n_{\theta'}^{T^t}-1)n_{\theta'}^{T^t}/2}\right)\right]^T\\
&\lg\left(1-\mathbf{p}\right)=\left[\lg\left(1-p_1\right),\lg\left(1-p_2\right),\ldots,\lg\left(1-p_{n_\varepsilon^{T^t}}\right)\right]^T.
\end{aligned} \tag{7}$$

In the process of inferring the congested link prior probabilities, the coefficient matrix of equations is sparse, and each element value is binary value of 0 or 1. Especially in larger scale IP network, when the pivot component $D_{d_{jj}}^{\prime\prime T^t}=0$ ($j=1,2,\ldots,n_\varepsilon^{T^t}$, $t=\{1,2\}$), we can not solve equation unique solutions. Even through $D_{d_{jj}}^{\prime\prime T^t}\neq 0$, $|D_{d_{jj}}^{\prime\prime T^t}|$ is also very small, and it will act as divisor in inversing operations; direct solution methods such as Gauss elimination method will cause a greater error because of rounding errors. Because coefficient matrix $D_d^{\prime\prime T^t}$ of prior probability solving equations is a typical sparse matrix in Boolean algebra model with less nonzero value, we used linear equation iterative algorithm to solve link prior probabilities [21].

4.3. Congested Link Inference Based on VSDDB Model. During the process of learning the link congested prior probabilities, we classify the different piece of time slice by the different routing matrix D in continuous snapshots. In continuous m times of snapshots, if the routing matrix D does not change, m times of snapshots can be divided into a time slice. According to Assumptions 4 and 5, we can simplify the solution process and only use the time slices T^1 and T^2 to construct simplified VSDDB network model. According to the present E2E path properties, we can infer the set of congested links by the following:

$$P\left(\mathbf{X}^{T^2}\mid\mathbf{Y}^{T^1},\mathbf{Y}^{T^2}\right)=\frac{P\left(\mathbf{X}^{T^2},\mathbf{Y}^{T^1},\mathbf{Y}^{T^2}\right)}{P\left(\mathbf{Y}^{T^1},\mathbf{Y}^{T^2}\right)}, \tag{8}$$

where $P(\mathbf{Y}^{T^1},\mathbf{Y}^{T^2})$ is only related to network state and path properties, which has nothing to do with selected links. So, the set of most likely congested links can be inferred by the following:

$$\begin{aligned}
&\arg\max P\left(\mathbf{X}^{T^2},\mathbf{Y}^{T^1},\mathbf{Y}^{T^2}\right)\\
&=\arg\max\sum_{j'=1}^{n_{\varepsilon'}} P\left(\mathbf{X}^{T^2},\mathbf{Y}^{T^1},\mathbf{Y}^{T^2}\mid\mathbf{X}_j^{T^1}\right)P\left(\mathbf{X}_j^{T^1}\right)\\
&=\arg\max\sum_{\mathbf{X}^{T^1}} P\left(\mathbf{X}^{T^1},\mathbf{X}^{T^2},\mathbf{Y}^{T^1},\mathbf{Y}^{T^2}\right)\\
&=\arg\max_{\mathbf{X}^{T^2}}\sum_{\mathbf{X}^{T^1}}\left\{P\left(x_{j'}^{T^1}\right)\cdot P\left[y_i^{T^1}\mid p_a\left(y_i^{T^1}\right)\right]\right.\\
&\left.\cdot P\left[y_i^{T^2}\mid p_a\left(y_i^{T^2}\right)\right]\cdot P\left(x_{j'}^{T^2},x_{j'}^{T^1}\right)\right\},
\end{aligned} \tag{9}$$

where $x_{j'}^{T^1}$ and $x_{j'}^{T^2}$, respectively, are state variables of node interface links in BNM of time slices T^1 and T^2. $n_{\varepsilon'}$ is the total number of node interface links. $p(y_i)$ is the parent nodes of y_i. In order to solve (9), we introduced Lemma 7.

Lemma 7. *$P[y_i\mid p_a(y_i)]$ get maximum value when meeting the following conditions: (1) if $y_i=0$ and $D_{ij}=1$, then, $x_j=0$; (2) if $y_i=1$, then, $x_j=1$ must exist; make $D_{ij}=1$.*

If status of some E2E path is good, all links traversed by this E2E path are good. Otherwise, at least having a link traversed by this E2E path is congested. So, there are probability relationships of (10) between some E2E path and corresponding traversing links.

$$P\left(y_i^{T^t} = 0 \mid p_a\left(y_i^{T^t}\right) = \{0, \ldots, 0\}\right) = 1,$$
$$P\left(y_i^{T^t} = 1 \mid \exists x_j^{T^t} = 1 \cap x_j^{T^t} \in p_a\left(y_i^{T^t}\right)\right) = 1. \tag{10}$$

According to (10), the maximum of $P[y_i^{T^t} \mid p_a(y_i^{T^t})]$ is equal to 1, because the state $x_{j'}$ of each link in IP network is a random variable with independent probability. Carrying out once snapshots to IP network in the present inference moment, the congestion probability inference distribution of each link obeys binomial probability equations of Bernoulli probability model $P_n(k) = C_n^k p^k (1-p)^{(n-k)}$, $n = 1$, so, the state probability of each link is independent. We can get the following:

$$P\left(x_{j'}\right) = \prod_{j'=1}^{n_{\varepsilon'}} p_j^{x_{j'}} \cdot \left(1 - p_{j'}\right)^{(1-x_{j'})}. \tag{11}$$

Because $P[y_i^{T^1} \mid p_a(y_i^{T^1})] = P[y_i^{T^2} \mid p_a(y_i^{T^2})] = 1$, (11) can be simplified as follows:

$$\arg\max P_p\left(X^{T^2} \mid Y^{T^1}, Y^{T^2}\right) = \arg\max_{\mathbf{X}^{T^2}} \sum_{X^{T^1}} P\left(x_{j'}^{T^1}\right) \cdot P\left(x_{j'}^{T^2}, x_{j'}^{T1}\right). \tag{12}$$

By Bayes theorem of $P(x_{j'}^{T^1}) \cdot P(x_{j'}^{T^2}, x_{j'}^{T^1}) = P(x_{j'}^{T^2}, x_{j'}^{T^1})$, because state variables of $x_{j'}^{T^1}$ and $x_{j'}^{T^2}$ are mutual independence, (12) can be expressed as follows:

$$\arg\max P\left(\mathbf{X}^{T^2} \mid \mathbf{Y}^{T^1}, \mathbf{Y}^{T^2}\right) = \arg\max_{\mathbf{X}^{T^2}} \sum_{\mathbf{X}^{T^1}} P\left(x_{j'}^{T^1}\right) \cdot P\left(x_{j'}^{T^2}\right). \tag{13}$$

$P(x_{j'}^{T^1})$ and $P(x_{j'}^{T^2})$, respectively, are congested prior probability of common links in time slices T^1 and T^2. Congested matrix in time slices T^1 and T^2 can be structured by corresponding congested E2E paths and corresponding traversing links. E2E path congestion probabilities of $P_i^{T^1}$ and $P_i^{T^2}$ in T^1 and T^2 can be calculated through taking average of congestion snapshots times in T^1 and T^2. Coefficient matrix of $D''^{T^1}_d$ and $D''^{T^2}_d$ in prior probability solving equations can be calculated by the congested matrix after getting rid of correlation and filling up full rank. We can, respectively, calculate the prior probabilities $p_{j'}^{T^1}$ and $p_{j'}^{T^2}$. Based on the Bernoulli experiment binomial distribution probability model, (13) can be translated into the following:

$$\arg\max_{\mathbf{X}^{T^2}} P\left(\mathbf{X}^{T^2} \mid \mathbf{Y}^{T^1}, \mathbf{Y}^{T^2}\right) = \arg\max_{\mathbf{X}^{T^2}} \prod_{j'=1}^{n_{\varepsilon''}} \left(p_{j'}^{T^1} \cdot p_{j'}^{T^2}\right)^{x_{j'}^{T^2}} \left(1 - p_{j'}^{T^1} \cdot p_{j'}^{T^2}\right)^{(1-x_{j'}^{T^2})}. \tag{14}$$

$n_{\varepsilon''}$ is total number in the present inference moment. After taking logarithm to (14), we can get the following:

$$\arg\max_{\mathbf{X}^{T^2}} P\left(\mathbf{X}^{T^2} \mid \mathbf{Y}^{T^1}, \mathbf{Y}^{T^2}\right) = \arg\max_{\mathbf{X}^{T^2}} \sum_{j'=1}^{n_{\varepsilon''}} \left[x_{j'}^{T^2} \lg\left(\frac{p_{j'}^{T^1} \cdot p_{j'}^{T^2}}{1 - p_{j'}^1 \cdot p_{j'}^2}\right) + \lg\left(1 - p_{j'}^{T^1} \cdot p_{j'}^{T^2}\right) \right], \tag{15}$$

where $\lg(1 - p_{j'}^1 \cdot p_{j'}^2)$ has nothing to do with link states; therefore, we can get the following:

$$\arg\max_{\mathbf{X}^{T^2}} P\left(\mathbf{X}^{T^2} \mid \mathbf{Y}^{T^1}, \mathbf{Y}^{T^2}\right) = \arg\max_{\mathbf{X}^{T^2}} \sum_{j'=1}^{n_{\varepsilon''}} x_{j'}^{T^2} \cdot \lg\left(\frac{p_{j'}^{T^1} \cdot p_{j'}^{T^2}}{1 - p_{j'}^1 \cdot p_{j'}^2}\right). \tag{16}$$

During the process of E2E path measurements, if routing of IP network never changes, (16) can be simplified into the inference method under the static routing IP network. That is to say, the congested link inference algorithm proposed in this paper is a kind of more general algorithm.

4.4. Congested Link Inference Algorithms. When the dimension of congested matrix is larger, existing optimal algorithms are hard to solve in the polynomial time. So, we proposed a kind of compromise method to solve the near-optimal solution in acceptable time to this Set Coverage Problem (SCP) [22]. We, respectively, proposed two kinds of algorithms: ICLINK based on the minimum set coverage method and CLILRS based on Lagrangian relaxation subgradient method, to infer the congested link feasible solutions.

To easily express the inference algorithms, we use p_k to express each of the link congested prior probabilities. If $t \geq 2$ (times of routing matrix, that is, the dynamic routing IP network), $p_k = p_{j'}^{T^1} \cdot p_{j'}^{T^2}$; if $t = 1$ (routing matrix does not change, that is, the static routing IP network), $p_k = p_j$. L_ε is the set of node interface links, $l_k \in L_\varepsilon$. $n(k)$ is the number of

(1) *Step 1*. Initial the set of congested links $\Omega = \phi$; initial the set of congested paths $P_\vartheta = P_\theta$;
(2) *Step 2*. In the inference moment, while ($P_\vartheta \neq \phi$)
(3) (1) Search link l_k: To some link l_k, when its value of $\lg[(p_k/(1-p_k)) \cdot n(k)]$ is the maximum value, $l_k \in L_\varepsilon$
(4) (2) Add l_k into the set Ω;
(5) (3)Renew the set of remaining congested link: Remove l_k from the remaining congested path;
(6) (4) Renew the set of E2E paths P_ϑ: Remove all paths contain the link l_k;
(7) (5) Return to line (2): *Step 2*;
(8) *Step 3*. Result outputs: The links in the set Ω are the inference results.

Algorithm 1: ICLINK algorithm.

links l_k traversed by E2E paths. So, (16) can be expressed as follows:

$$\begin{aligned} &\arg\max_{\mathbf{x}^{T^2}} P\left(X^{T^2} \mid Y^{T^1}, Y^{T^2}\right) \\ &= \arg\max_{\mathbf{x}^{T^2}} \sum_{k=1}^{n_{\varepsilon''}} x_k \cdot \lg\left(\frac{p_k}{1-p_k}\right). \end{aligned} \tag{17}$$

The pseudocode of ICLINK is listed in Algorithm 1.

The congested link inference algorithms are a $\lg(n_{\varepsilon''}+1)$ approximation algorithm with a computational complexity of $O(n_{\varepsilon''} n_{\theta'})$. But, because ICLINK is still based on the minimum coverage set criterion, when IP network shows multiple link congestion, the inference performance of ICLINK will sharply descend in theory.

According to (17), we conclude the following based on the Lagrangian relaxation theory [23] to solving this SCP.

$$\begin{aligned} &z_{sc} = \min \quad \sum_{j=1}^{n_{\varepsilon''}} c_j x_j, \quad c_j = -\lg\frac{p_k}{1-p_k} \\ &\text{s.t.} \quad \sum_{j=1}^{n_{\varepsilon''}} D'''_{d_{ij}}, \quad x_j \geq 1, \ i = 1, 2, \ldots, n_{\theta'} \\ &\qquad x_j \in \{0, 1\}, \quad j = 1, 2, \ldots, n_{\varepsilon''}, \end{aligned} \tag{18}$$

where $D'''_{d_{ij}}$ is obtained through removing good paths and corresponding traversing links from $D''^{T^2}_{d}$. We can relax (18) into following optimization problem and obtain

$$\begin{aligned} z_{\mathrm{LRSC}}(\mathbf{u}) = \min &\sum_{j=1}^{n_{\varepsilon''}} d_j x_j + \sum_{i=1}^{n_{\theta'}} u_i, \\ &x_j \in \{0, 1\}, \ j = 1, 2, \ldots, n'_\varepsilon, \end{aligned} \tag{19}$$

where $d_j = c_j - \sum_{i=1}^{n_{\theta'}} u_i D'''_{d_{ij}}$ is the Lagrangian multiplier, $u_i \geq 0$ ($i = 1, 2, \ldots, n_{\theta'}$).

The value of z_{LD} expresses the infimum algorithm searched. Z_{UB} denotes the optimal feasible solution of (18). Z_{LB} denotes some feasible solution of (19). The pseudocode of CLILRS is listed in Algorithm 2.

5. Experimental Evaluation

Experimental evaluation methods in IP network mainly have three kinds: analog experiments, simulation experiments, and actual Internet experiments. Among them, standard answers are known in analog experiment; experimental details can be completely controlled, but the disadvantage is not real enough. Actual network experiments have the true environment, but the standard answer is hard to obtain. So, in this paper, we carried out three kinds of experiments to evaluate the performance of algorithms.

5.1. Analog Experimental Evaluation. In order to simulate the actual network scenarios, we adopt network topology generator Brite [24] to simulate IP network topology models of Waxman, BA, and GLP [2] and evaluate the algorithm performance in the different type and scale by using the random number model to simulate multiple link congested events. We carried out evaluations on the Eclipse platform.

5.1.1. Selecting Value N in the Learning Process. Under the learning stage of link congested prior probability, in order to assure inference performance and reduce cost, selecting suitable value of N is very important. In order to select optimal value of N, we carried out two groups of experiments of the new algorithm ICLINK under the different scenarios.

(1) Different Node Number ($f = 0.2$). We produced the topology model of Waxman, BA, and GLP in node number = 50, 100, and 300 by using topology generator Brite. Through changing N under selecting $f = 0.2$, we can, respectively, get E2E path measurement results. By adopting formula (7), we can get link congested prior probabilities and then, based on the algorithm of ICLINK, we can infer the congested link positions. Under the three kinds of topology models, the results of DR and FPR by ICLINK algorithm, respectively, are listed in Figure 3.

From Figure 3 we can see that, under the different network scale, each link congestion law can be learned after selecting about 30 times' snapshots.

(2) Different f (Node Number = 100). We produced the topology model of Waxman, BA, and GLP in node number = 100 by using Brite topology generator. Through changing N under selecting $f = 0.1$, 0.3, and 0.5, we can, respectively, get E2E path measurement results. By adopting formula (7),

(1) *Step 1*. Initial $z_{\mathrm{LD}} = -\infty$, $z_{\mathrm{UB}} = \infty$, $u_i = \min[c_j | D'''_{d_{ij}} = 1,\ \ j = 1, 2, \ldots, n_{\varepsilon''}]$, $i = 1, 2, \ldots, n_{\theta'}$;
(2) *Step 2*. Use the present $\mathbf{u}$ to solve equations (18), and make the optimal-value z_{LB}, renew $z_{\mathrm{LD}} = \max(z_{\mathrm{LD}}, z_{\mathrm{LB}})$;
(3) *Step 3*. Construct a feasible solution of the initial SCP:
(4) (1) Make $S = [j | x_j = 1, 2, \ldots, n_{\varepsilon''}]$;
(5) (2) To uncovered path P_i ($\sum_{j=1}^{n_{\varepsilon''}} D'''_{d_{ij}} x_j = 0$), add $\min[j \mid D'''_{d_{ij}} = 1,\ d_j < \infty,\ \ j = 1, 2, \ldots, n_{\varepsilon''}]$ corresponding j to S;
(6) (3) Arrange $j \in S$ in descending order, If $S - [j]$ is still a feasible solution of SCP, $S = S - [j]$;
(7) (4) Renew $z_{\mathrm{UB}} = \min(z_{\mathrm{UB}}, \sum_{j \in S} c_j)$.
(8) *Step 4*. $z_{\mathrm{LD}} = \max z_{\mathrm{LRSC}}(\mathbf{u})$, If $z_{\mathrm{LD}} = z_{\mathrm{UB}}$, Return to line (13): *Step 9*;
(9) *Step 5*. Compute sub-gradient $g_i = 1 - \sum_{j=1}^{n_{\varepsilon''}} D'''_{d_{ij}} x_j$, $i = 1, 2, \ldots, n_{\theta'}$. If $\sum_{i=1}^{n_{\theta'}} (g_i)^2 = 0$, return to line (13): *Step 9*;
(10) *Step 6*. Compute iteration step-length $\delta = f(1.05 z_{\mathrm{UB}} - z_{\mathrm{LB}}) / \sum_{i=1}^{n_{\theta'}} (g_i)^2$, the initial value $f(0) = 2$,
If z_{LD} do not add in continuous 30 times of iterations, then f is halved;
(11) *Step 7*. If $f \leq 0.005$, return to line (13): *Step 9*;
(12) *Step 8*. Renew Lagrangian multiplier $u_i = \max[0, u_i + \delta g_i]$, $i = 1, 2, \ldots, n_{\theta'}$, return to line (2): *Step 2*;
(13) *Step 9*. Output $L_j = \{l_j \mid x_j = 1\}$ is the optimal solution.

Algorithm 2: CLILRS algorithm.

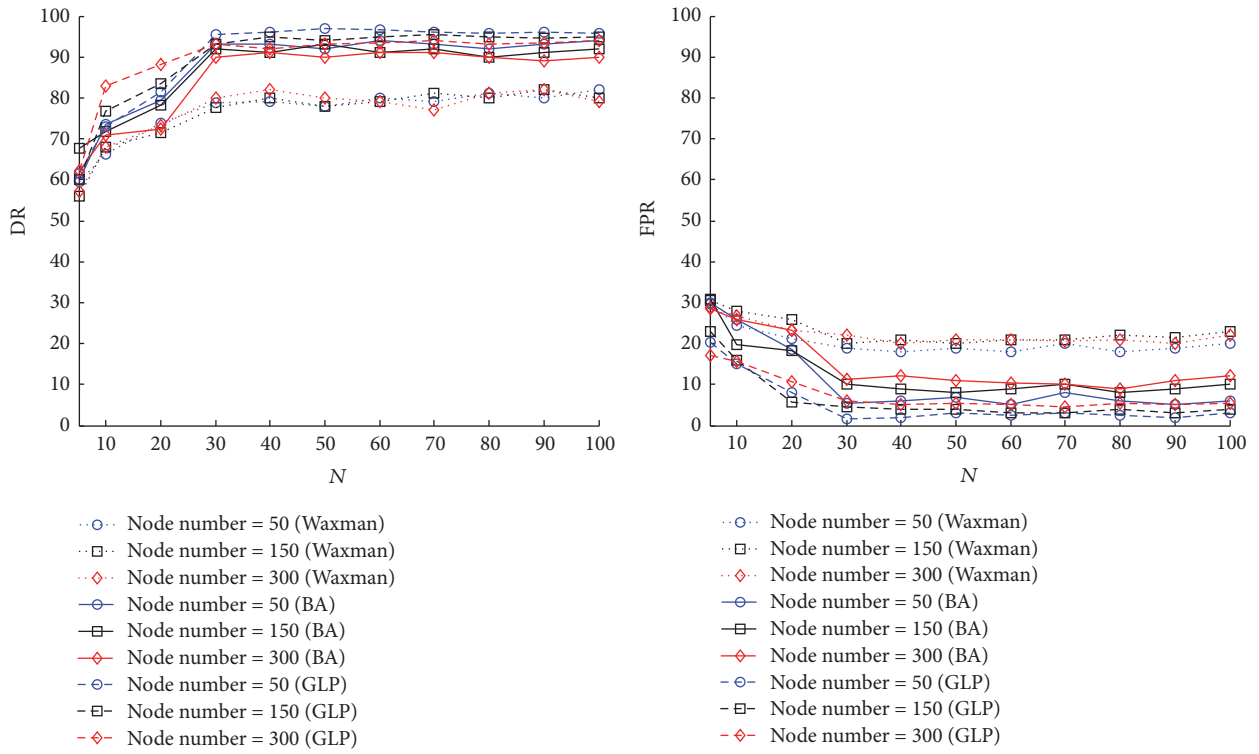

Figure 3: DRs and FPRs of ICLINK algorithm under different values of N (under different network scale).

we can get link congested prior probabilities, and then, based on the algorithm of ICLINK, we can infer the congested link positions. Under the three kinds of topology models, the results of DR and FPR by ICLINK algorithm, respectively, are listed in Figure 4.

From Figure 4 we can see that, under the different network topology, each link congestion law can be learned through more than 30 times' snapshots. So, we treat $N = 30$ as the snapshots' times in the process of prior probability learning stage. The prior probability solving process of each link is not introduced in the manuscript any longer.

5.1.2. Parameter Setting and Evaluation Index

Path-Congested-Threshold. E2E paths are good when N number of congested snapshots is less than *path-congested-threshold*. Default setting $N = 30$; *path-congested-threshold* = 0.

Congested-Link-Ratio. In order to prove algorithm performance in different congested links ratio of experiments, we introduce a parameter *congested-link-ratio*; set $0.05 \leq$ *congestion-link-ratio* ≤ 0.6 to assume the number of congested links.

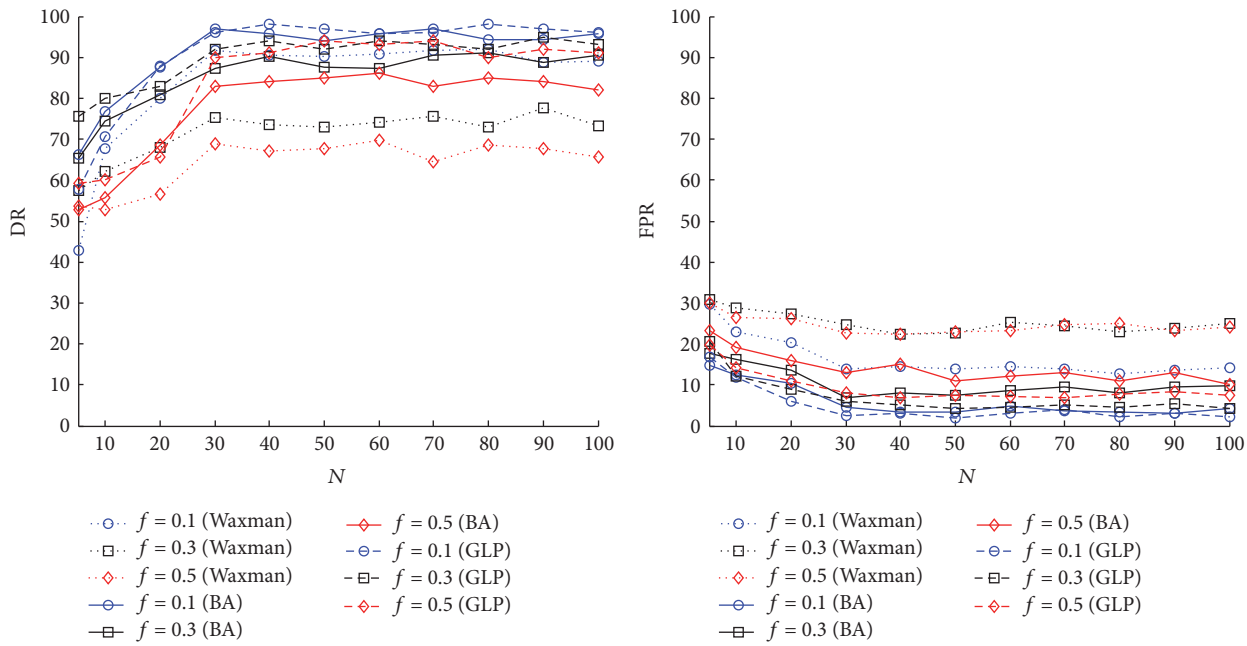

Figure 4: DRs and FPRs of ICLINK algorithm under different values of N (under different congested-link-ratio f).

Route-Changing-Threshold. In order to prove performance impacts in the different routing changing frequency of IP network, we introduce parameter *route-changing-threshold* as the condition of changing-routing; when the number of link continuous congestions reaches r*oute-changing-threshold*, E2E path will reroute. The default value of *route-changing-threshold* = 4.

In order to prove performance impacts in different changing-routing frequencies, we use Detection Rate (DR) and False Positive Rate (FPR) list as follows to evaluate algorithm performance and take the average of continuous 10 times of experiment results under invariable threshold parameters.

$$\begin{aligned} \mathrm{DR} &= \frac{F \cap X}{F} \\ \mathrm{FPR} &= \frac{X \setminus F}{X}, \end{aligned} \tag{20}$$

where F is the set of actual congested links (benchmark) in the present inference moment. X is the set of links inferred by a local algorithm. From the definition we can see that the calculating result of DR is higher, the inference results more conform to the actual benchmark of congested links, the calculating result of FPR is lower, and the erroneous judgments of congested links are fewer. So, higher DR and lower FPR of some congested link inference algorithm express that the inference performance is better than others.

5.1.3. Inference Performance under Static Routing IP Network

(1) Performance Impact in the Different Congested-Link-Ratio. Under the different network models of Waxman, BA, and GLP, if the routing does not change, inference process is the same as CLINK. So, we only compare the inference performance of ICLINK and CLILRS proposed in this paper; results are shown in Figure 5.

We observed that CLILRS is significantly more accurate than ICLINK under the different IP network models. Better performance of CLILRS boils down to the one fact that it does not use the theory of minimum set coverage. And for this reason, CLILRS also does not have lower FPRs than ICLINK, but the difference is little. With the increase of *congested-link-ratio*, DRs of two algorithms both present descending tendency. DRs of CLILRS are higher than ICLINK under the different topology network models. DRs are the highest under GLP model; models of BA and Waxman followed. This is directly related to the topology structure; Waxman model has longer path length, but models of BA and GLP are power-law models; parts of routers have bigger degree value.

(2) Performance Impact in the Different Network Scales. To prove the inference performance of two algorithms proposed in this paper under the different IP network scales, we carry out congested link inference under *node number* = [50, 500], *congestion-link-ratio* = 0.5. Inference results are shown in Figure 6.

Under the different types and scales of network, with the increase of node number, the inference performance of ICLINK and CLILRS both present the slow descending tendency. The inference performance of CLILRS under the different network types is all superior to ICLINK. DRs under the GLP model are the highest; models of BA and Waxman followed. FPRs under the GLP model are the lowest; models of BA and Waxman followed. With the increase of the network scale, FPRs of ICLINK and CLILRS under three

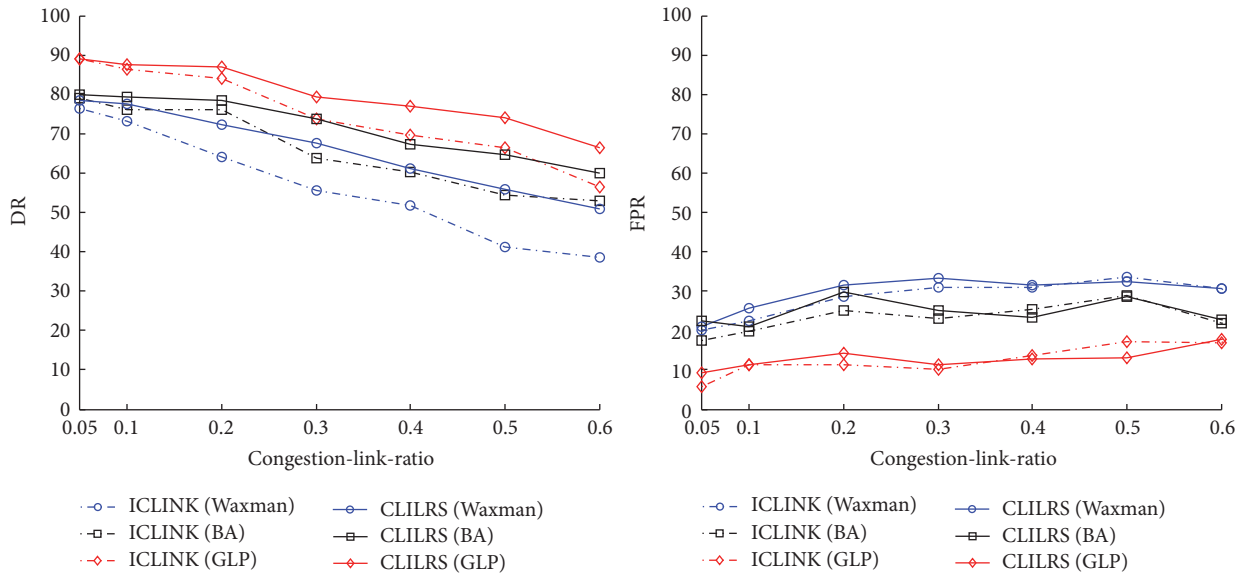

Figure 5: DRs and FPRs of CLILRS and ICLINK under three kinds of different topologies and *congested-link-ratio* changing from 0.05 to 0.6 (*node number* = 150).

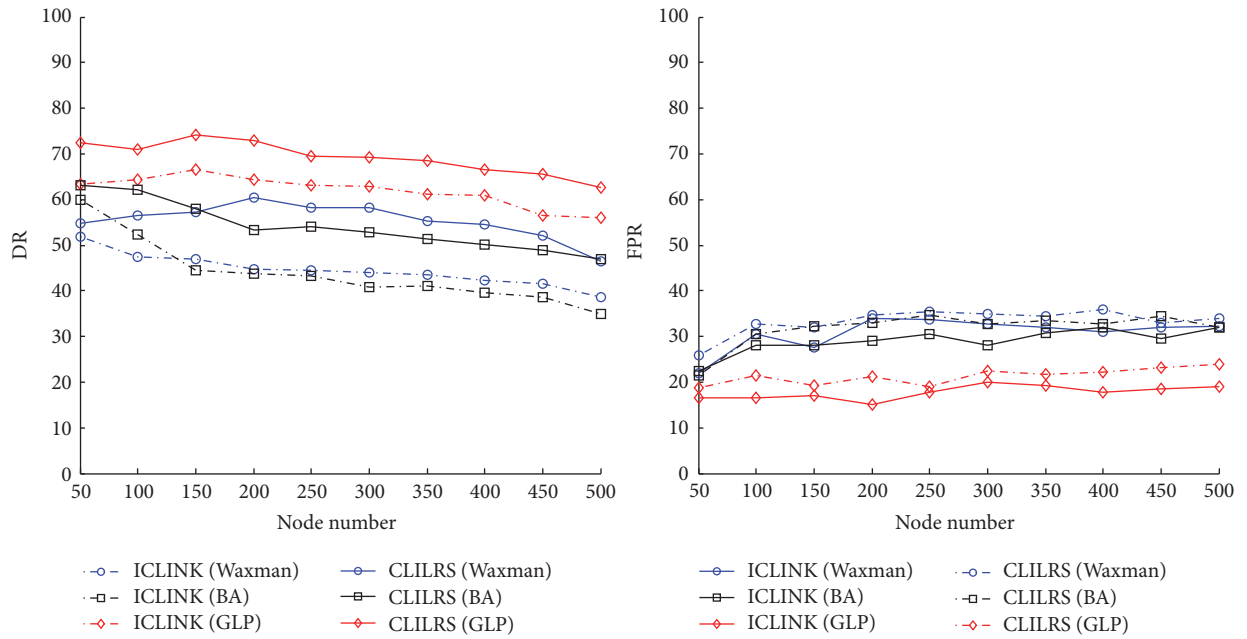

Figure 6: DRs and FPRs of CLILRS and ICLINK under three kinds of different topologies and different network scales (*congestion-link-ratio* = 0.5).

kinds of models all remain stable. Among them, FPRs under the GLP model are the lowest; models of BA and Waxman followed. FPRs of two algorithms have little difference.

(3) Performance Impact under the Different Link Coverage. When selecting the different routers as end nodes of E2E paths, we can get the different link coverage. Under the different link coverage, we, respectively, use ICLINK and CLILRS to infer the set of congested links. Under 150 nodes of IP network model with the default setting of Brite, the degree value of topology network models Waxman, BA, and GLP, respectively, is 2~14, 2~33, and 1~33. We select the E2E path according to the shortest path first strategy under the different router *degree value*. Under the Waxman model, when selecting router *degree value* = 8~10 and 12~14, the E2E paths and link coverage are unchanged. Under the BA model, when *degree value* = 12~15 and 16~33, E2E paths and link coverage are unchanged. Under the GLP model, when *degree value* = 5~7 and 9~33, the E2E paths and link coverage are unchanged. So, in order to express inference performance

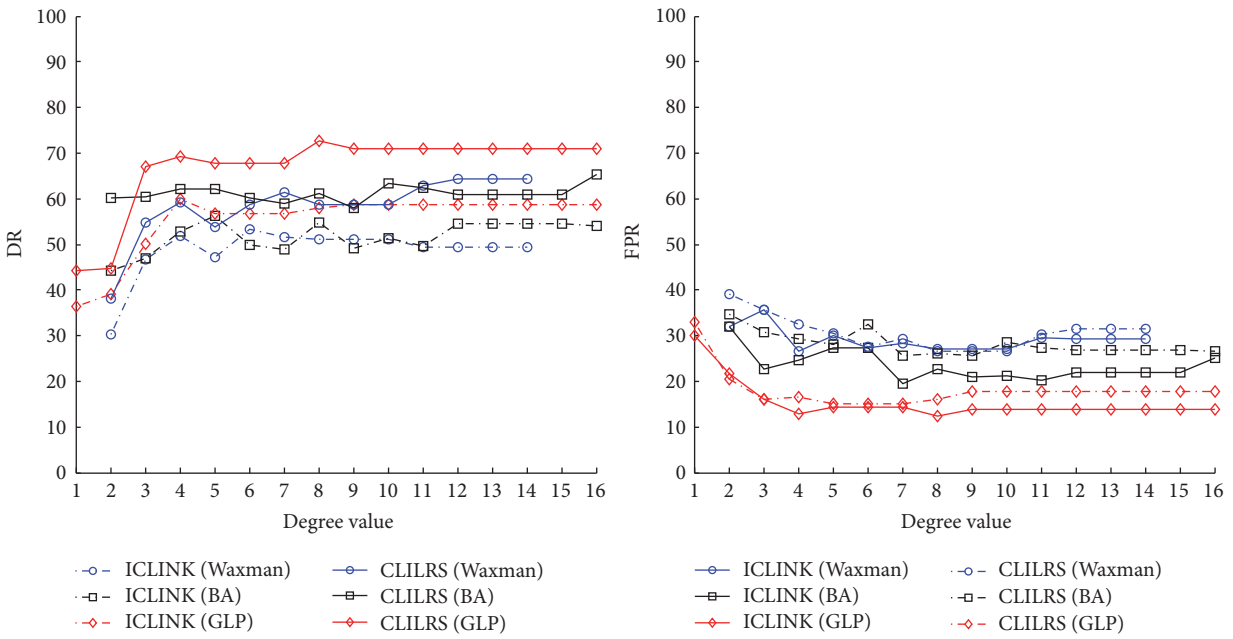

Figure 7: DRs and FPRs of CLILRS and ICLINK under three kinds of different topologies (*node number* = 150, *congested-link-ratio* = 0.5).

under the different *degree value* in the same coordinate axis, we select *degree value* of 1~16 as the horizontal axis. Performance of ICLINK and CLILRS under the multiple link congested scenarios (*congestion-link-ratio* = 0.5) is shown in Figure 7.

When selecting leaf node (end host node) with minimum *degree value* in IP network, because the link coverage under models of GLP and Waxman is insufficient, DRs are obviously descending and with higher FPRs. But under the BA model with 150 nodes, because the link coverage is enough under *degree value* = 2, the inference performance is not affected. But, when the *degree value* selection reaches certain values, the inference performance of ICLINK and CLILRS tends to be stable and the inference performance of CLILRS is averagely higher above 10% than ICLINK.

5.1.4. Inference Performance under Dynamic Routing IP Network. In order to evaluate the effectiveness and accuracy of algorithms CLINK, ICLINK, and CLILRS in dynamic routing IP network, we simulate the link congested events in the different *congested-link-ratio* based on the random number model. We assume the *route-changing-threshold* (default value of route-changing-threshold is 4) as the times of changing-routing when the link was continuous in the congested state. Based on the strategy of the shortest path first, E2E paths will reroute. We, respectively, use the algorithms of CLINK, ICLINK, and CLILRS to infer the set of congested links in the present inference moment.

(1) Performance Impact in the Different Congested-Link-Ratio. Under the different network models, we repeat each simulation 10 times and set *congested-link-ratio* = [0.05, 0.6]; results are shown in Figure 8.

We observed that CLILRS and ICLINK are significantly more accurate than ICLINK with higher DRs and FPRs under the different IP network models. Better inference performance of CLILRS and ICLINK boils down to the main reason that two algorithms are based on VSDDB model proposed in this paper. DRs of CLILRS and ICLINK are the highest under the GLP model; models of BA and Waxman followed. Under the dynamic routing IP network, the inference performance of CLINK obviously reduces a lot compared to static routing IP network. Because the GLP model has stronger power-law, impacts of E2E paths traversing links are obvious; the descending of DRs is 50% higher than static routing IP network. Under models of BA and Waxman, performance averagely descends about 40% and 30%. Under dynamic routing IP network, DRs of CLIRLS and ICLINK averagely descend about 10% compared to static routing IP network. FPRs averagely rise 10% compared to static routing IP network. With the increase of *congested-link-ratio*, CLILRS and ICLINK show stronger robustness; stability performance is obviously superior to CLINK.

(2) Performance Impact under the Different Routing Changing Frequencies. In order to prove the inference performance under the different IP network topology models, we simulate the *link continuous congestion times* to trigger routing changing as the changing-routing frequency in IP network. During experiments, we, respectively, set *link continuous congestion times* from 1 to 4. That is, routing changes when *link continuous congestion times* = 1, which expresses that changing-routing is frequently in IP network, *link continuous congestion times* = 4 otherwise.

Under 150 nodes' IP network (*congested-link-ratio* = 0.1), we, respectively, use different algorithms of CLINK, ICLINK,

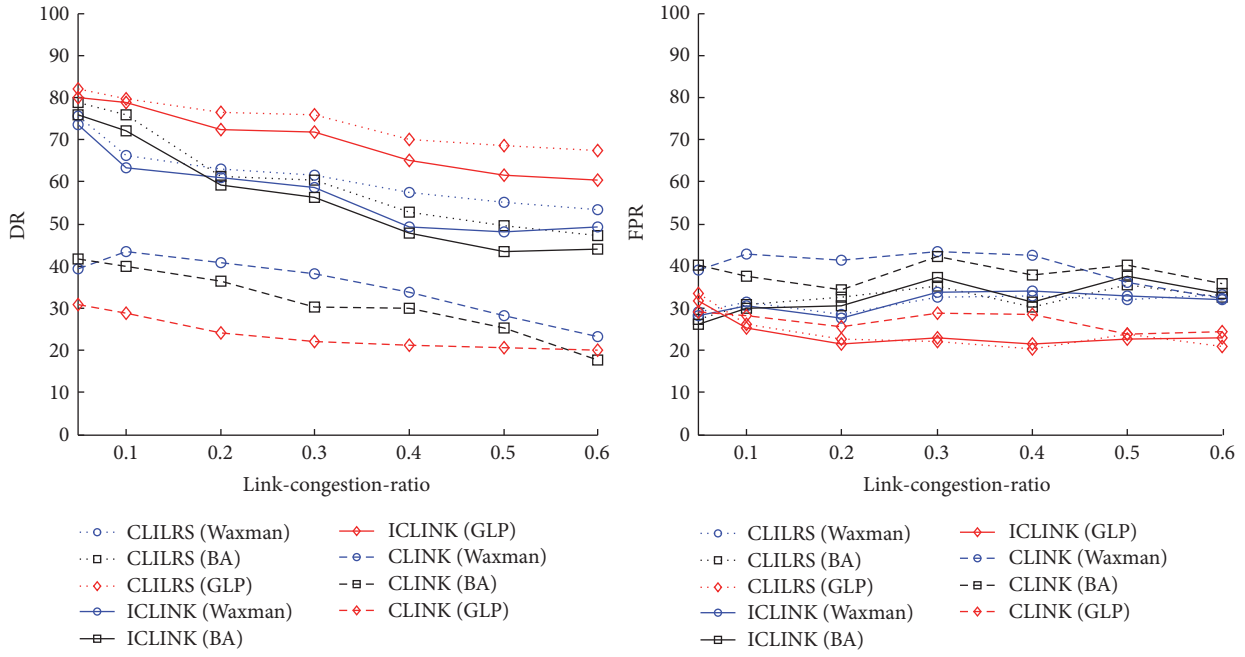

Figure 8: DRs and FPRs of CLILRS, ICLINK, and CLINK under different *congested-link-ratio* (*node number* = 150).

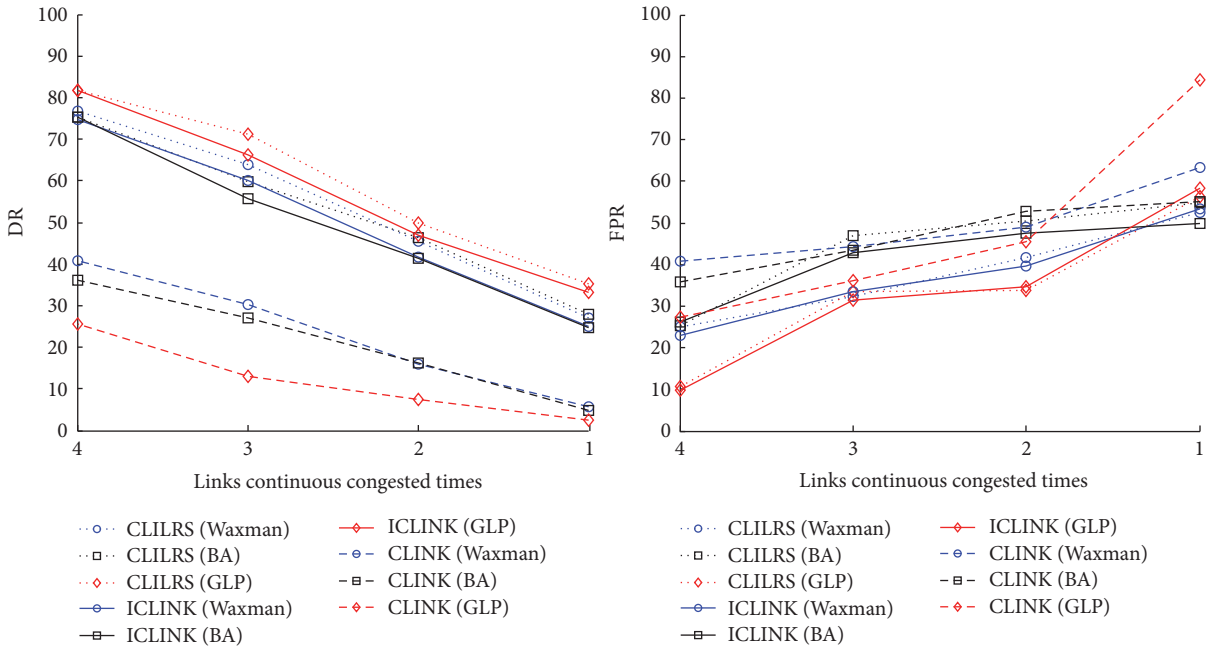

Figure 9: DRs and FPRs of CLILRS, ICLINK, and CLINK under the different topologies and different routing changing frequencies (*congested-link-ratio* = 0.1).

and CLIRLS to infer the set of congested links; DRs and FPRs are shown in Figure 9.

The routing changing is faster, DR is lower, and FPR becomes higher. With the increasing of changing-routing frequency, DR under different network topology, respectively, presents linear descending tendency. Under the GLP model, DRs of CLIRLS and ICLINK are both the highest, FRPs are both the lowest, and models of BA and Waxman followed. Under the different network topologies, performances of CLIRLS and ICLINK are all superior to CLINK.

(3) Performance Impact under the Different Network Scales. In order to prove the inference performance under the different network scales, under the models of Waxman, BA, and GLP

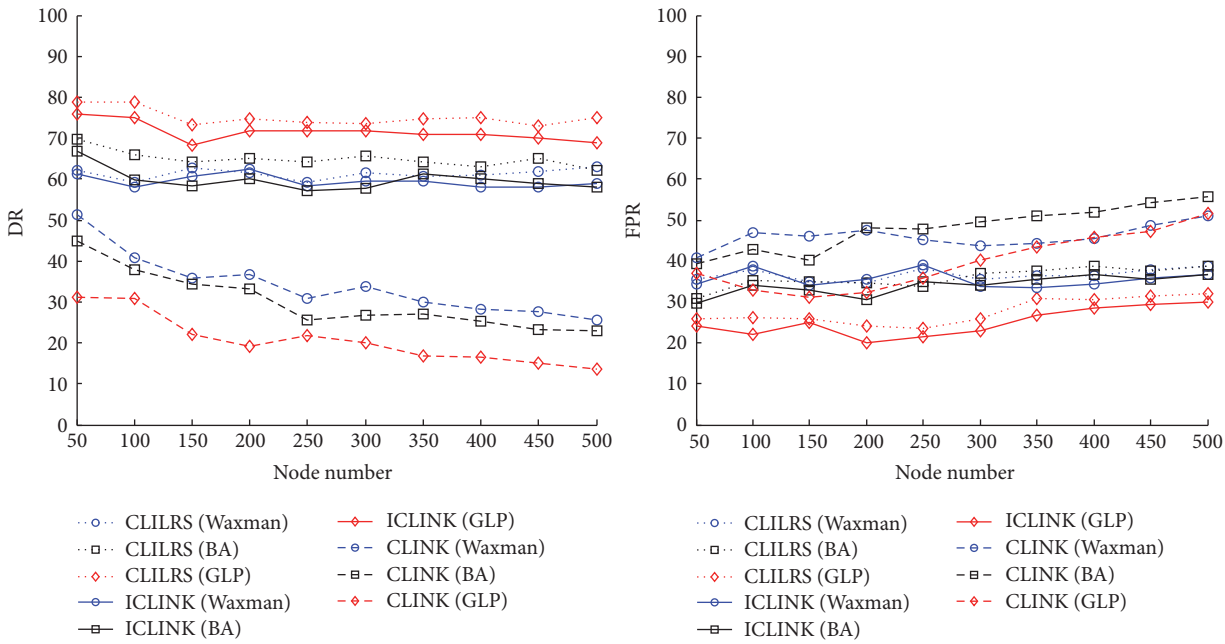

Figure 10: DRs and FPRs of CLILRS, ICLINK, and CLINK under the different topology and network scale (*congested-link-ratio* = 0.2).

(*node number* = [50, 350], *congested-link-ratio* = 0.2, and *route-changing-threshold* = 4), DRs and FPRs of CLINK, ICLINK, and CLILRS are shown in Figure 10.

Under the different IP network models, the inference performances of CLILRS and ICLINK are both superior to CLINK. The inference performance of CLIRLS is superior to CLINK, CLIRLS and ICLINK both have the highest DR under the GLP model, and models of BA and Waxman followed. FPRs of CLINK, ICLINK, and CLIRLS are the lowest under the GLP model; models of BA and Waxman followed. To the different scale of IP network, the inference performance of CLIRLS and ICLINK has higher robustness; DRs and FPRs are more stable than CLINK.

(4) Performance Impact in the Different Link Coverage. In order to evaluate the performance impacts of CLIRLS and ICLINK under the different links coverage, we simulate 150 nodes' IP network models of Waxman, BA, and GLP and, respectively, set router degree value from 2 to 14 as probe deployment router to cover links of IP network. Under the different IP network models, we, respectively, use three kinds of algorithms to infer the set of congested links. DRs and FPRs are shown in Figure 11.

When *degree-threshold-value* >3, the impact of inference performance for CLINK, ICLINK, and CLIRLS is little. DRs and FPRs of CLIRLS and ICLINK both have higher stability. The changes of DRs and FPRs are within 5%. When selecting the minimum degree value (Waxman = 2, BA = 2, and GLP = 1) to obtain E2E paths, the inference performance of ICLINK and CLILRS under models of Waxman and GLP obviously descends because the link coverage in IP network is not enough.

DRs of CLIRLS and ICLINK are both the highest under the GLP model, and models of BA and Waxman followed. The inference performance of CLIRLS is superior to ICLINK. FPRs under the GLP model is the lowest; models of BA and Waxman followed. DRs of CLINK under three kinds of models are all lower than 50%. FPRs of CLIRLS and ICLINK are all lower than CLINK under three kinds of models. FPRs all are the lowest under the GLP model; models of BA and Waxman followed. Under the different IP network models, FPRs of CLILRS and ICLINK are all lower than CLINK, FPRs of CLILRS and ICLINK are the lowest under the GLP model, and models of BA and Waxman followed. Under the different network models, FPRs of CLILRS and ICLINK are, respectively, lower than CLINK. FPRs of CLILRS and ICLINK are the lowest under the GLP model and models of BA and Waxman followed. FPRs of CLILRS and ICLINK are, respectively, lower at least 10% than CLINK.

5.2. Simulation Experimental Evaluation. We carried out simulation experiments on Emulab platform under the power-law model GLP; simulation experiment scenarios were designed as follows: (1) use Brite topology generator to produce 20-node GLP network model; (2) import Brite file, build up a test network topology, and join up probes and performance monitor console into waiting test network; (3) we transmit measurement tasks to each node probe to measure E2E path properties and obtain paths and corresponding traversing links in each time slice of snapshots. Measurement information will be uploaded to performance monitor. We, respectively, use local algorithms to evaluate inference performance.

5.2.1. Constructing Simulation Experiment Scenarios. (1) Read the topology file into Emulab simulation experiment platform and set each link bandwidth of 100 Mbps and time

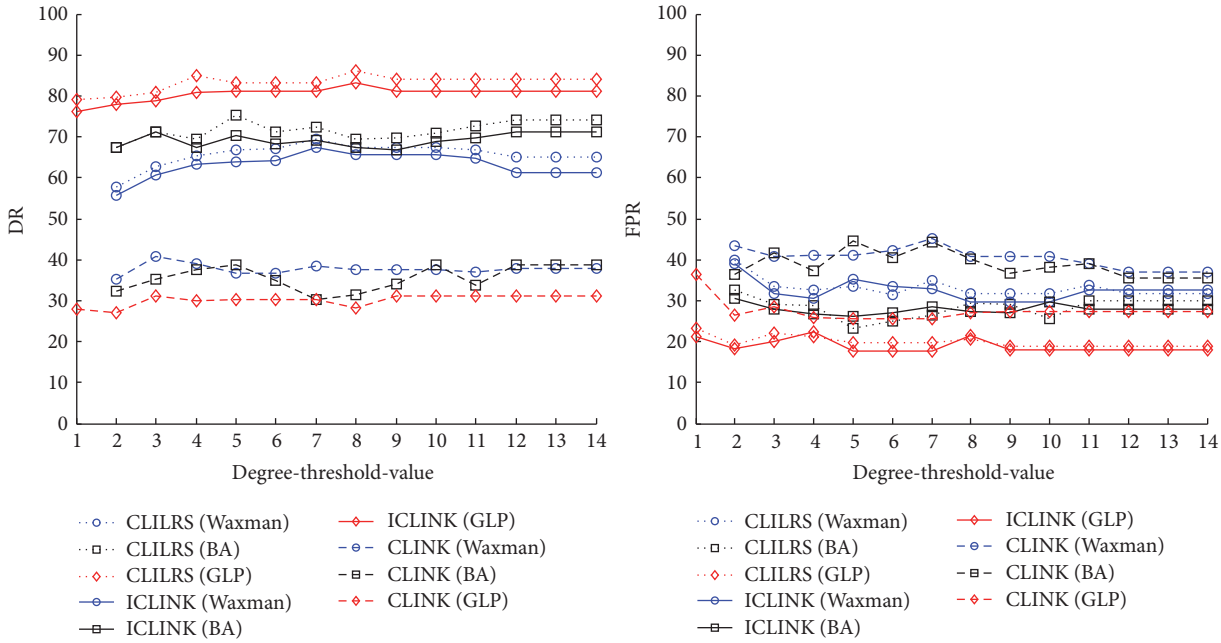

Figure 11: DRs and FPRs of CLILRS, ICLINK, and CLINK under different *degree value* of routers (*node number* = 150, *congested-link-ratio* = 0.1).

delay of 15 ms. (2) Set OSPF protocol in test IP network and carry out E2E path snapshots every 2 min. Each E2E path follows the rule of shortest path first. E2E path will reroute after link congestion time delay beyond 8 min. Each routing table can realize stability within 2 minutes, and all of links can resume normal data transmission within 2 min. (3) In IP network using LM1 model [18], when some link is congested, loss rate obeys uniform distribution in the range of [0.05, 1]. When some link is normal, loss rate obeys uniform distributed in the range of [0, 0.01]. After initializing loss rate for each link, link loss rate will follow Gilbert process. That is, each link state fluctuates between the state of congestion and normal. When some link lies in normal state, there is no packets loss; all packets are lost otherwise. And when some link loss rate is greater than 0.05, the link is congested. Otherwise, the link loss rate is less than 0.01 in normal. Every 2 min, IP network will generate random congestion links by the *congested-link-ratio*. The topology network model of Emulab simulation experiment is listed in Figure 12.

5.2.2. Simulation Experimental Results. To prove the inference performance under Emulab simulation experiment platform, we simulate *congested-link-ratio* = [0, 0.6] to produce the different proportion of link congested events and, respectively, use CLINK, ICLINK, and CLILRS to infer the set of congested links in the present inference moment. We repeat each simulation 10 times and report average DR and FPR in Figure 13.

From Figure 13 we can see the inference performance of CLIRLS and ICLINK proposed in this paper is obviously higher than CLINK. We set *congested-link-ratio* = 0.5 and *route-changing-threshold* = 4 during the simulation experiment, because of dynamic routing protocol OSPF; in some simulation experiments, there are twice routing changing during simulation experiments, and the link coverage has certain changes in the two time slices T^1 and T^2. Links of 2|0, 1|10, and 1|5 in T^1 did not appear in T^2. And links of 1|0 and 3|5 in T^2 did not appear in T^1. The congested link inference DR of CLIRLS, ICLINK, and CLINK is, respectively, 81%, 75%, and 62.5%, and FPR, respectively, is 5%, 0, and 16.7%.

5.3. Actual Internet Experimental Evaluation. In order to evaluate the inference performance of algorithms proposed in this paper in actual IP network, we carried out experiments in an actual IP network. We first use traceroute to measure the links traversed by E2E paths to construct the routing matrix in each time slice. Traceroute is performed sequentially from each vantage point to all other vantage points. Similarly to the simulations, in the actual Internet experiments, we take 30 snapshots to estimate the prior probabilities of links. That is, we use the snapshots obtained in the previous two hours to learn the prior. We then use these probabilities to infer congested links for the next measurement snapshots of the network. The following results are the average results of 10 experiments, and each is separated by 4 hours from the previous one. To prevent overloading the nodes, each host sends probes to the other hosts in a random order.

Because link loss rate in actual network is hard to obtain, there is no benchmark (F) in (20). We use the agreement rate of path properties [13] to evaluate algorithm inference performance.

(1) We divided E2E path set into two equal sets in a random way, that is, inference path set of **I** and validation path set of **P**. (2) In the path set of **I**, we carried out snapshots every 4 min in each E2E path, total 30 times. We used simple

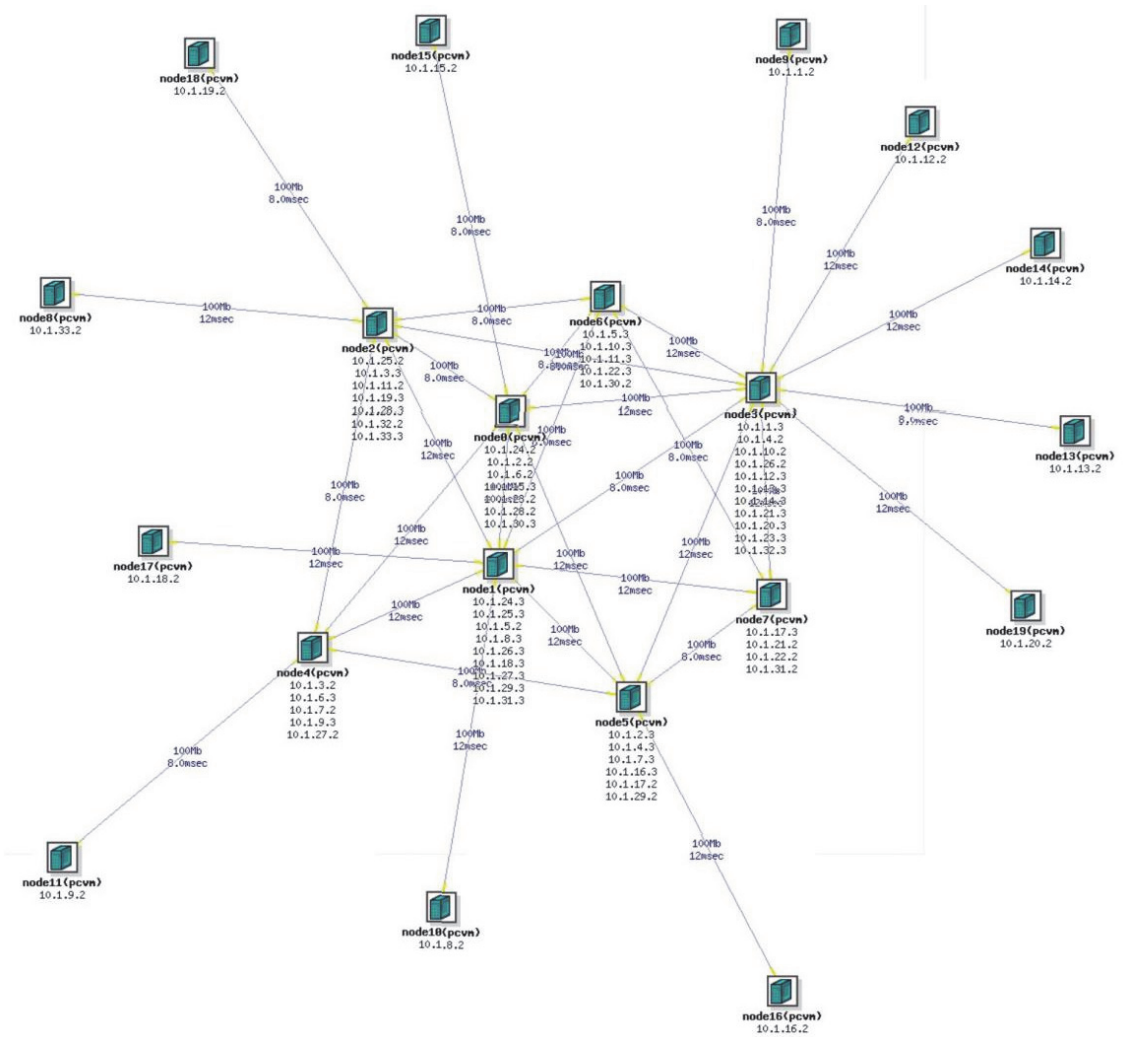

Figure 12: Topology network of Emulab simulation experiment.

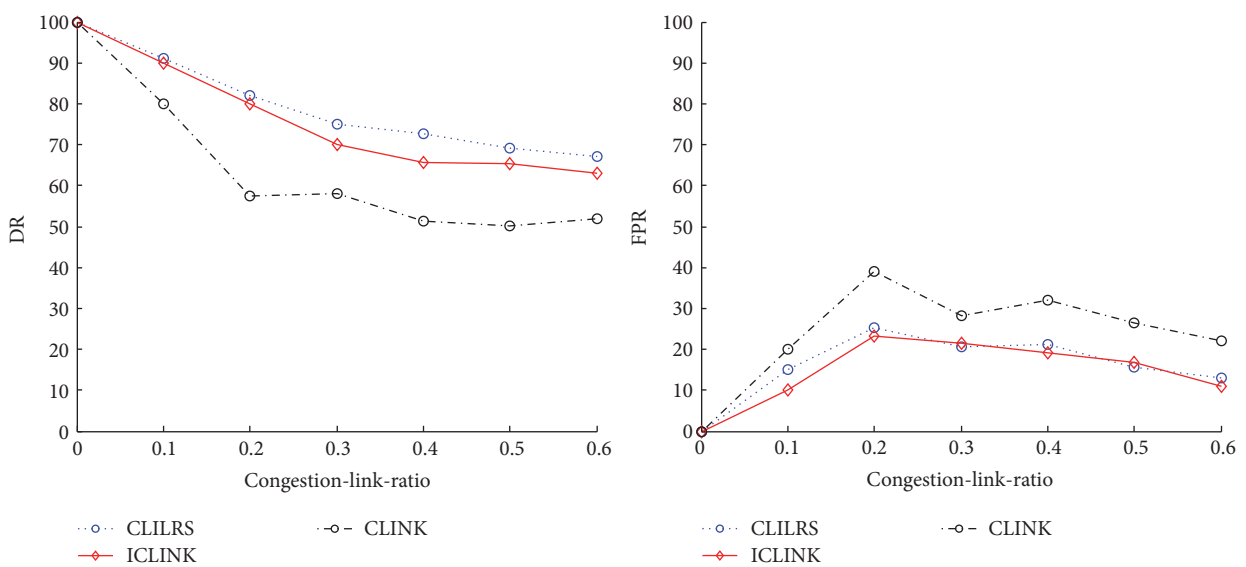

Figure 13: Emulab simulation experimental results.

Table 1: Actual IP network experiments.

Algorithm	Agreement rate of path properties
CLINK	79.3%
ICLINK	90.1%
CLIRLS	93.6%

UDP packets as probes. Each host sends 100 UDP packets of size of 60 bytes to every other host. Time between probes follows an exponential distribution with a mean value of 0.2 seconds. According to E2E path measurement results, we can get each link prior probabilities in the set of **I**. (3) Through carrying out once snapshots in all E2E paths to get each traversing links and path properties, we can infer the congested links by algorithms of CLIRLS, ICLINK, and CLINK in the path set of **I**. (4) According to the rule that a congested path at least has a congested link, we can judge congested path in the set of **P** according to results obtained by the set of **I**. (5) Through contrasting the actual path loss rate in the present inference moment, we can, respectively, evaluate the properties of each algorithm by the agreement rate of path properties:

$$\text{agreement rate of path properties} = \frac{\text{agreement number of link properties}}{\text{total link number}}. \quad (21)$$

Inference performance comparisons in actual IP network experiments are shown in Table 1.

Because CLINK algorithm does not fully consider the dynamic routing in Internet, links traversed by E2E paths will change when the link bandwidth is limited, and so on; inference accuracy of algorithm CLINK is lower than ICLINK and CLIRLS.

6. Conclusions

Aiming at dynamic routing IP network, a kind of simplified VSDDB network model was built through introducing assumptions of Markov property and time homogeneity. According to descending of inference performance in dynamic routing IP network, algorithms of ICLINK and CLILRS were proposed. Experiments proved accuracy and robustness of algorithms proposed in this paper. Inference performances of CLILRS and ICLINK are all much better than CLINK under dynamic routing IP network. In addition, inference performance of CLILRS is obviously better than ICLINK under the multiple link congested scenarios.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this article.

Acknowledgments

This study was supported by the Aviation Science Foundation Project of China (2014ZD55007 and 2014ZD55010).

References

[1] C. Orsini, E. Gregori, L. Lenzini, and D. Krioukov, "Evolution of the Internet *k*-dense structure," *IEEE/ACM Transactions on Networking*, vol. 22, no. 6, pp. 1769–1780, 2014.

[2] H. X. Nguyen and P. Thiran, "The boolean solution to the congested IP link location problem: theory and practice," in *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, pp. 2117–2125, IEEE, Anchorage, Alaska, USA, May 2007.

[3] Y. Qiao, X. S. Qiu, and L. Cheng, "Active probing based method for fault diagnosis using Bayesian network," *China Communications*, vol. 8, no. 7, pp. 1–11, 2011.

[4] V. Paxson, "Growth trends in wide-area TCP connections," *IEEE Network*, vol. 8, no. 4, pp. 8–17, 1994.

[5] F. Qian and G. M. Hu, "A survey of network tomography technologies," *Computer Science*, vol. 33, no. 9, pp. 12–16, 2006.

[6] S. Zarifzadeh, M. Gowdagere, and C. Dovrolis, "Range tomography: combining the practicality of boolean tomography with the resolution of analog tomography," in *Proceedings of the ACM Internet Measurement Conference (IMC '12)*, pp. 385–398, Boston, Mass, USA, November 2012.

[7] Y. Vardi, "Network tomography: estimating source-destination traffic intensities from link data," *Journal of the American Statistical Association*, vol. 91, no. 433, pp. 365–377, 1996.

[8] A. Adams, T. Bu, T. Friedman et al., "The use of end-to-end multicast measurements for characterizing internal network behavior," *IEEE Communications Magazine*, vol. 38, no. 5, pp. 152–159, 2000.

[9] T. Bu, N. Duffield, F. L. Presti, and D. Towsley, "Network tomography on general topologies," in *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pp. 21–30, Marina Del Rey, Calif, USA, June 2002.

[10] Y. Y. Mao, F. R. Kschischang, B. H. Li, and S. Pasupathy, "A factor graph approach to link loss monitoring in wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 820–829, 2005.

[11] M. Coates, A. O. Hero III, R. Nowak, and B. Yu, "Internet tomography," *IEEE Signal Processing Magazine*, vol. 19, no. 3, pp. 47–65, 2002.

[12] A. Batsakis, T. Malik, and A. Terzis, "Practical passive lossy link inference," in *Passive and Active Network Measurement: 6th International Workshop, PAM 2005, Boston, MA, USA, March 31- April 1, 2005. Proceedings*, vol. 3431 of *Lecture Notes in Computer Science*, pp. 362–367, Springer, Berlin, Germany, 2005.

[13] N. G. Duffield, "Network tomography of binary network performance characteristics," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5373–5388, 2006.

[14] P. Yi, K. Qian, W.-W. Huang, J. Wang, and Z. Zhang, "Adaptive flow sampling algorithm based on sampled packets and force sampling threshold S towards anomaly detection," *Journal of Electronics & Information Technology*, vol. 37, no. 7, pp. 1606–1611, 2015.

[15] E. Lawrence, G. Michailidis, V. N. Nair, and B. Xi, "Network tomography: a review and recent developments," in *Frontiers in Statistics*, pp. 345–364, Imperial College Press, 2006.

[16] S. L. Pan, Z. Y. Zhang, G. L. Fei, F. Qian, and G. M. Hu, "Survey on network tomography for link performance parameter evaluation," *Journal of Software*, vol. 26, no. 9, pp. 2356–2372, 2015.

[17] Y. Gu, G. F. Jiang, V. Singh, and Y. P. Zhang, "Optimal probing for unicast network delay tomography," in *Proceedings of the IEEE INFOCOM*, pp. 1–9, IEEE, San Diego, Calif, USA, March 2010.

[18] V. N. Padmanabhan, L. Qiu, and H. J. Wang, "Server-based inference of internet performance," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies*, vol. 4, pp. 1–15, San Francisco Calif, USA, April 2003.

[19] X. G. Gao and J. G. Shi, "Structure varied discrete dynamic Bayesian network and its inference algorithm," *Journal of Systems Engineering*, vol. 22, no. 1, pp. 9–14, 2007.

[20] Y. Shavitt, X. Sun, A. Wool, and B. Yener, "Computing the unmeasured: an algebraic approach to internet mapping," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 1, pp. 67–78, 2004.

[21] Y. Vardi and D. Lee, "From image deblurring to optimal investments: maximum likelihood solutions for positive linear inverse problems," *Journal of the Royal Statistical Society, Series B: Methodological*, vol. 55, no. 3, pp. 569–612, 1993.

[22] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, New York, NY, USA, 1993.

[23] M. Shakeri, K. R. Pattipati, V. Raghavan, and A. Patterson-Hine, "Optimal and near-optimal algorithms for multiple fault diagnosis with unreliable tests," *IEEE Transactions on Systems, Man and Cybernetics Part C*, vol. 28, no. 3, pp. 431–440, 1998.

[24] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: an approach to universal topology generation," in *Proceedings of the 9th International Symposium in Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '01)*, pp. 346–353, August 2001.