

# 层析成像技术中的自适应网络拓扑推断算法

赵 洪 华, 陈 鸣, 魏 镇 韩

(解放军理工大学 指挥自动化学院, 江苏 南京 210007)

**摘要:** 基于层析成像技术的网络拓扑推断由于采用单一的网络性能参数而在推断时受到网络负载的影响. 为了减少单一参数在推断时的限制, 提出了一种单播网络中自适应结合多种网络性能参数的拓扑推断算法, 该算法能适应不同网络负载的环境并且在应用时仅需适量的计算而无需增加测量流量. 分析了自适应拓扑推断算法的正确性并通过 NS2 进行了仿真, 仿真结果表明自适应网络拓扑推断算法能在不同网络负载的环境下准确的推断网络拓扑.

**关键词:** 网络层析成像; 拓扑推断; 自适应

**中图分类号:** TP393    **文献标识码:** A    **文章编号:** 1001-2400(2009)03-0547-06

## Self adaptive network topology inference algorithm in tomography

ZHAO Hong-hua, CHEN Ming, WEI Zhen-han

(Institute of Command Automation, PLA Univ. of Sci. & Tech., Nanjing 210007, China)

**Abstract:** There are a few network topology inference techniques based on network tomography, but all of them use only one network performance characteristics, which leads to many limits when the network load is different. In order to reduce the limits of the inference based on one network performance characteristics, a self adaptive network topology inference method is proposed which joins multiple network performance parameters self-adaptively in inference. In applying the self adaptive network topology inference, no additional traffic is needed except for some calculation, and the inference method could be applied in complex networks with different loads. The self adaptive method is analyzed theoretically and validated through simulations by NS2, and the results of simulation illustrate that the self-adaptive inference method could infer network topology correctly when faced with networks whose load changes greatly.

**Key Words:** network tomography; topology inference; self adaptive

网络层析成像<sup>[1-2]</sup>可以通过端到端的技术来推断网络内部的特性, 而不需要网络内部节点的合作, 例如在网络拓扑已知的情况下可以通过端到端的时延或端到端的丢包率推断内部链路的时延或丢包率<sup>[3-4]</sup>. 网络拓扑推断是网络层析成像技术的最新应用之一, 它根据网络中节点性能特性的相关性来推断网络拓扑结构. 研究表明<sup>[5]</sup>网络中节点的共享链路越多, 节点的特性越相近, 即相关性越大.

目前基于网络层析成像技术的拓扑推断技术大都采用单一的端到端性能参数来推断网络的拓扑结构<sup>[6-9]</sup>, 基于单一性能参数拓扑推断在计算时较简单, 但也存在很多限制, 例如不同网络性能参数适应的网络负载不同. 为了减少基于单一性能参数在拓扑推断时受网络负载的影响, 笔者提出了一种自适应的网络拓扑推断算法, 该算法仅需适量的计算而无需增加测量流量.

## 1 相关研究

基于层析成像的拓扑推断主要分为两个步骤: ①通过端到端的测量获得测量源节点到测量目标节点的

端到端性能参数,根据端到端性能参数计算出测量目标节点间的相关性;②根据测量目标节点间的相关性推断网络的拓扑结构。

在基于层析成像技术的拓扑推断中,节点端到端性能的测量需要采用特殊的方式,比较常用的测量方法主要有两种:紧接(back to back)分组对<sup>[3-4]</sup>方法和“三明治”分组列车<sup>[6]</sup>方法,这两种方法的基本思想是相邻的分组在共享链路具有相同的网络特性。在测量时,每次测量两个网络节点,以获得两个节点间的相关性。通过紧接分组对方法可以测量节点的端到端丢包率和端到端时延,并且端到端丢包率和端到端时延可以计算节点间的相关性,通过“三明治”分组列车可以测量的节点的性能参数是排队时延,排队时延即可作为节点间的相关性参数。

推断算法主要包括:DBT 算法<sup>[5]</sup>,MLT<sup>[6]</sup>算法,LBT 算法<sup>[7]</sup>和分级算法<sup>[8]</sup>等,DBT 算法根据节点的相关性推断网络的拓扑结构,该算法采用自底向上的推断,推断的拓扑结构为二叉树形的网络拓扑结构;MLT 是基于最大似然估计的全局搜索算法,该算法在所有可能的拓扑树集合中搜索最优的拓扑树,该算法的计算量比较大,尤其是当节点数较多时计算量非常大;LBT 和分层算法都是基于聚类的拓扑推断算法,该算法按节点的相关性进行聚类,直到生成一棵完整的树。

上述拓扑推断算法都是基于单一网络性能参数计算的节点相关性来推断网络拓扑,因此在网络负载变化较大时推断的效果较差。

## 2 理论基础

拓扑推断中的测量有 3 个假定条件:①空间独立性,即数据分组在不同链路上是相互独立的;②时间独立性,即同一链路上不同数据分组是相互独立的。③网络拓扑在测量过程中是固定不变的。

在网络拓扑推断中涉及如下两个定义。

**定义 1** 节点间的相关性值是一种在树状结构的网络拓扑中,反映叶节点在网络中共享链路特性的值,叶节点的共享链路越多相关性值越大,反之亦然。

**定义 2** 设  $T = (V, L)$  表示树状拓扑,其中  $V$  为节点集, $L$  为链路集, $o$  表示根节点同时为测量源节点, $R$  为叶节点集同时为测量目标节点集, $R = \{1, 2, \dots, i, \dots, j\}$ 。令  $a(i, j)$  为叶节点  $i$  和叶节点  $j$  的父节点, $n_i < n_j$  表示  $n_i$  是  $n_j$  的子孙。如果  $a(k, j) < a(i, j)$ ,则节点  $k$  和  $j$  的共享链路比节点  $i$  和  $j$  的共享链路多。

### 2.1 时延协方差

在单播网络中,时延协方差可通过紧接分组对的方法测量并计算得到,紧接分组对由两个大小相同的分组组成,分组之间具有较小的时间间隔,两个分组分别到达不同的目标地址。测量时由测量源节点向测量目标节点对发送紧接分组对,根据在目标节点观察到的端到端时延计算出两个目标节点间的时延协方差,时延协方差作为节点间的相关性值。

如图 1,设紧接分组对的目的地址为叶节点  $i$  和叶节点  $j$ ,称为叶节点对  $(i, j)$ 。令  $a(i, j)$  为叶节点对  $(i, j)$  的父节点, $\mathbf{Y}_i^{(i,j)}, \mathbf{Y}_j^{(i,j)}$  分别表示从根节点  $S$  到达叶节点  $i$  和叶节点  $j$  的端到端时延向量, $\mathbf{X}_a^{(i,j)}, \mathbf{X}_i^{(i,j)}, \mathbf{X}_j^{(i,j)}$  分别表示链路  $(S \rightarrow a(i, j))$ 、链路  $(a(i, j) \rightarrow i)$  和链路  $(a(i, j) \rightarrow j)$  的时延向量。则从根节点  $S$  到叶节点  $i$  和叶节点  $j$  的时延协方差<sup>[8]</sup>为

$$\text{Cov}(\mathbf{Y}_i^{(i,j)}, \mathbf{Y}_j^{(i,j)}) = \text{Cov}(\mathbf{X}_a^{(i,j)} + \mathbf{X}_i^{(i,j)}, \mathbf{X}_a^{(i,j)} + \mathbf{X}_j^{(i,j)}) = \text{Var}(\mathbf{X}_a^{(i,j)}) \quad (1)$$

如图 1 所示,根节点  $S$  到内部节点  $a(i, j)$  的路径有两部分  $S \rightarrow c(i, j)$  和  $(c(i, j) \rightarrow a(i, j))$  组成,则

$$\text{Var}(\mathbf{X}_a^{(i,j)}) = \text{Var}(\mathbf{X}_1^{(i,j)} + \mathbf{X}_2^{(i,j)}) = \text{Var}(\mathbf{X}_1^{(i,j)}) + \text{Var}(\mathbf{X}_2^{(i,j)}) \quad (2)$$

把图 1 的情况转为一般化,设链路  $(S \rightarrow a(i, j))$  由链路  $L_1, L_2, \dots, L_n$  组成, $\mathbf{X}_1^{(i,j)}, \mathbf{X}_2^{(i,j)}, \dots, \mathbf{X}_n^{(i,j)}$  为链路  $L_1, L_2, \dots, L_n$  的时延,根据条件 1,各链路相互独立,则

$$\text{Var}(\mathbf{X}_a^{(i,j)}) = \text{Var}(\mathbf{X}_1^{(i,j)} + \mathbf{X}_2^{(i,j)} + \dots + \mathbf{X}_n^{(i,j)}) = \text{Var}(\mathbf{X}_1^{(i,j)}) + \text{Var}(\mathbf{X}_2^{(i,j)}) + \dots + \text{Var}(\mathbf{X}_n^{(i,j)}) \quad (3)$$

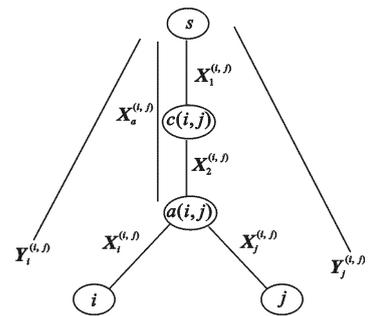


图 1 时延协方差计算

由式(1)和式(3)可知,在树状网络拓扑中叶节点对的共享链路越多,叶节点对的时延协方差越大,因为叶节点对的时延协方差(共享链路的时延方差)是加性的关系。

当网络负载适中时测量分组在共享链路的时延变化较大,节点间共享链路的时延方差较大.在网络负载较轻时,测量分组在共享链路的时延变化较小,而在网络负载较重时丢包较多,影响到紧接分组对的测量,所以时延协方差在网络负载适中时能准确地反映节点共享链路的情况,推断网络拓扑的正确性最高。

### 2.2 成功传输率

单播网络中成功传输率可以通过紧接分组对的方法测量并计算得到,由根节点向所有叶节点对发送紧接分组对,根据叶节点收到测量分组的数量,计算丢包率和成功传输率,用成功传输率作为叶节点间的相关性值。

如图 2 所示,设  $P_a^{(i,j)}$ ,  $P_i^{(i,j)}$ ,  $P_j^{(i,j)}$  分别表示分组在链路( $S \rightarrow a(i,j)$ )、链路( $a(i,j) \rightarrow i$ )和链路( $a(i,j) \rightarrow j$ )中成功传输的概率,而  $1 - P_a^{(i,j)}$ ,  $1 - P_i^{(i,j)}$ ,  $1 - P_j^{(i,j)}$  分别表示相应链路的丢包率。

设  $S_i^{(i,j)}$  为叶节点  $i$  收到测量分组的概率,  $S_j^{(i,j)}$  为叶节点  $j$  收到测量分组的概率.  $S_{i,j}^{(i,j)}$  为在叶节点  $i$  和叶节点  $j$  收到测量分组对的概率. 由图 2 可知,叶节点  $i$  和叶节点  $j$  共享链路的成功传输率<sup>[8]</sup>为

$$P_a^{(i,j)} = S_i^{(i,j)} \times S_j^{(i,j)} / S_{i,j}^{(i,j)} \quad (4)$$

设链路( $S \rightarrow a(i,j)$ )由链路  $L_1, L_2, \dots, L_n$  组成,  $P_1^{(i,j)}, P_2^{(i,j)}, \dots, P_n^{(i,j)}$  分别为链路  $L_1, L_2, \dots, L_n$  的成功传输率,则

$$P_a^{(i,j)} = P_1^{(i,j)} \times P_2^{(i,j)} \times \dots \times P_n^{(i,j)} \quad (5)$$

由于  $P_x^{(i,j)} \leq 1$ ,所以在树状网络拓扑中叶节点对的共享链路越多,则成功传输率  $P_a^{(i,j)}$  越小,因为  $P_a^{(i,j)}$  是乘性的关系。

基于成功传输率的拓扑推断在网络负载较重时效果最好,而在网络负载较轻时由于丢包率较小,成功传输率不能准确地反映节点间共享链路的情况。

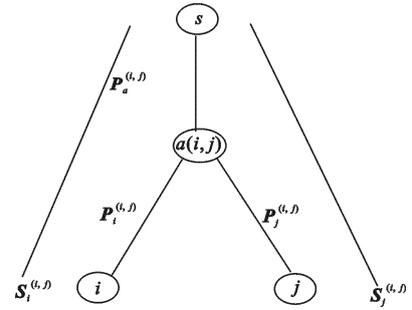


图 2 成功传输率计算

## 3 自适应网络推断算法

自适应网络推断算法的基本思想是通过紧接分组对测量方法同时测量时延和丢包率,然后计算时延协方差和成功传输率,采用时延协方差和成功传输率相结合的值作为拓扑推断的节点对相关性值。

当网络负载适中时,时延协方差作为节点对相关性值对节点共享链路的反映较好,而丢包率很小,因此在推断中,采用的拓扑推断参数主要依据时延协方差;当网络负载较重时,丢包率较高,导致计算时延协方差的紧接分组对样本数减少,通过少量样本计算的协方差与真实值偏离较大,因此基于时延协方差的方式在网络负载较重时误判的概率较大,而成功传输率更能反映节点间共享链路的情况,因此在网络负载较重时采用的拓扑推断参数主要依据成功传输率。

### 3.1 参数转换

在基于层析成像技术的拓扑推断中,结合成功传输率和时延协方差则可以适用于网络负载适中和网络负载较重的情况,而且时延协方差和成功传输率可以通过紧接分组对的测量方法同时获得。

根据式(3)和式(5),时延协方差在链路中是加性的关系,而成功传输率是乘性的关系,并且成功传输率  $P \leq 1$ ,两个参数没有相同的规律,因此对两个参数作相应的转换,然后采用加权求和的方式计算新的节点间相关性值。

首先对成功传输率作对数计算,把乘性关系转换为加性关系,然后把计算的结果取反,则成功传输率和时延协方差在节点间共享链路上具有相同的递增特性,在后面的推断中成功传输率都是指经过转换后的成功传输率.成功传输率的转换计算如下,令  $L_a^{(i,j)} = -\log(P_a^{(i,j)})$ , 则

$$\begin{aligned} L_a^{(i,j)} &= -\log(P_a^{(i,j)}) = -\log(S_i^{(i,j)} \times S_j^{(i,j)} / S_{i,j}^{(i,j)}) \Rightarrow L_a^{(i,j)} = \\ &= -[\log(S_i^{(i,j)}) + \log(S_j^{(i,j)}) - \log(S_{i,j}^{(i,j)})] \Rightarrow L_a^{(i,j)} = \log(S_{i,j}^{(i,j)}) - \log(S_i^{(i,j)}) - \log(S_j^{(i,j)}) \quad (6) \end{aligned}$$

令  $T_a^{(i,j)}$  表示叶节点对  $(i,j)$  的时延协方差,  $L_a^{(i,j)}$  表示叶节点对  $(i,j)$  的成功传输率, 根据式(3), 式(5)和式(6)可得到下面推理.

$$\begin{aligned} (T_a^{(k,j)} + L_a^{(k,j)}) - (T_a^{(i,j)} + L_a^{(i,j)}) &> (L_a^{(k,j)} - L_a^{(i,j)}) \quad ; \\ (T_a^{(k,j)} + L_a^{(k,j)}) - (T_a^{(i,j)} + L_a^{(i,j)}) &> (T_a^{(k,j)} - T_a^{(i,j)}) \quad . \end{aligned} \quad (7)$$

通过推理可以发现, 时延协方差  $T$  与转换后的成功传输率  $L$  的相加运算对节点间共享链路的特性起到放大作用. 由于不同参数的量纲不同, 简单的把参数相加并不合适, 为了解决不同参数量纲采用下面的方法标准化, 然后对时延协方差和成功传输率加权求和.

### 3.2 参数标准化

设每个叶节点对  $(i,j)$  有  $N$  组相关性样本值, 其中叶节点对  $(i,j)$  的时延协方差样本为  $(\hat{T}_1^{(i,j)}, \hat{T}_2^{(i,j)}, \dots, \hat{T}_N^{(i,j)})$ , 叶节点对  $(i,j)$  的成功传输率样本为  $(\hat{L}_1^{(i,j)}, \hat{L}_2^{(i,j)}, \dots, \hat{L}_N^{(i,j)})$ , 均值计算如下

$$\bar{T}_{i,j} = \sum_{k=1}^N \hat{T}_k^{(i,j)} / N, \quad \bar{L}_{i,j} = \sum_{k=1}^N \hat{L}_k^{(i,j)} / N \quad . \quad (8)$$

通过式(8)计算节点对  $(i,j)$  的平均时延协方差和平均成功传输率, 作为节点对  $(i,j)$  的时延协方差值和成功传输率值.

设拓扑树  $T$  中有  $M$  个叶节点, 则共有  $\binom{M}{2}$  个样本协方差值和样本成功传输率值, 令  $T$  和  $S$  分别为节点对的样本协方差和转换的样本成功传输率的集合, 则  $T = \{\bar{T}_{1,2}, \bar{T}_{1,3}, \dots, \bar{T}_{i,j}, \dots, \bar{T}_{m-1,m}\}$ ,  $S = \{\bar{S}_{1,2}, \bar{S}_{1,3}, \dots, \bar{S}_{i,j}, \dots, \bar{S}_{m-1,m}\}$ , 令  $L$  为节点对的丢包率集合, 则  $L = \{\bar{L}_{1,2}, \bar{L}_{1,3}, \dots, \bar{L}_{i,j}, \dots, \bar{L}_{m-1,m}\}$ ,  $E(L)$  为样本丢包率的均值. 设  $D(T), D(S)$  分别为  $T$  和  $S$  的方差,  $\min(T)$  和  $\min(S)$  分别为  $T$  和  $S$  最小值, 对  $T$  和  $S$  线性标准化计算, 标准化公式为

$$\bar{T}_{i,j} = (\bar{T}_{i,j} - \min(T)) / (D(T))^{1/2}, \quad \bar{S}_{i,j} = (\bar{S}_{i,j} - \min(S)) / (D(S))^{1/2} \quad . \quad (9)$$

对于所有叶节点对  $(i,j)$ , 采用自适应参数  $J_{i,j}$  表示两个节点间的相关性值,  $J = \{\bar{J}_{1,2}, \bar{J}_{1,3}, \dots, \bar{J}_{i,j}, \dots, \bar{J}_{m-1,m}\}$ ,  $\bar{J}_{i,j}$  的计算公式为

$$\bar{J}_{i,j} = (1 - (E(L))^{1/2}) \times \bar{T}_{i,j} + (E(L))^{1/2} \times \bar{S}_{i,j} \quad . \quad (10)$$

采用自适应参数与采用时延协方差和成功传输率相比增加了计算量, 增加的计算量主要在 3 个方面: ①成功传输率转换的计算, 计算复杂度为  $O(n)$ ; ②对于时延协方差和成功传输率的标准化计算, 计算复杂度为  $O(n)$ ; ③自适应参数的计算, 计算复杂度为  $O(n)$ . 因此采用自适应参数与原有采用时延协方差和采用成功传输率的算法相比, 增加的计算量为  $O(n) + O(n) + O(n)$ , 计算复杂度为  $O(n)$ . 由于时延和丢包率在测量过程中可以同时得到, 因此与原有算法相比采用自适应计算不会增加测量流量.

### 3.3 拓扑推断算法 (TPIA)

由 3.1 和 3.2 节的计算公式可知, 在树状网络拓扑中, 叶节点  $i$  和叶节点  $j$  的共享链路越多,  $\bar{J}_{i,j}$  越大. 推断算法则是根据  $\bar{J}_{i,j}$  的值来推断节点间共享链路的情况, 从而推断网络拓扑, 拓扑推断算法如下:

输入: 根节点  $\{0\}$ , 叶节点集  $V_r = \{\{1\}, \dots, \{M\}\}$ , 边集  $L = \emptyset$ ,  $J = \{\bar{J}_{1,2}, \bar{J}_{1,3}, \dots, \bar{J}_{i,j}, \dots, \bar{J}_{m-1,m}\}$

输出:  $T = (V, L)$

While  $|V_r| > 1$  do

Begin

根据  $J$  对节点集划分  $V_r = A_1 \cup \dots \cup A_i \dots \cup A_n$ , 划分的原则是: 对于每一个集合  $A_i$ ,

$$\forall i \forall j (i, j \in A_i) \rightarrow \bar{J}_{i,j} > \bar{J}_{i,m} (\forall m \notin A_i);$$

Foreach ( $A_i \subset V_r$ ) do

Begin

If  $A_i$  内节点数  $> 1$  then

Begin

合并  $A_i$  内的节点, 生成新的节点  $m$ ;

更新  $T$ , 在  $T$  中减去  $A_i$  中节点的相关性值, 并加入新值  $\bar{J}_{m,k} (k \notin A_i) = \max(\bar{J}_{i,k} (i \in A_i))$ ;

更新节点集  $V$ , 加入新节点  $m, V = A_i \cup \{m\}$ ;  
 更新边集  $L, L = L \cup \{(m, x), x \in A_i\}$ ;  
 更新叶节点集  $V_r, V_r = \{V_r - A_i\} \cup \{m\}$ ;

End;

End;

End.

## 4 NS2 仿真

为了验证自适应拓扑推断算法的有效性,在 NS2<sup>[10]</sup> 环境下进行了仿真实验,并在不同网络负载的情况下验证了算法的有效性.

仿真实验采用较为普遍的树状结构的网络拓扑.由于选路协议的作用,基于单个发送节点,多个接收节点的拓扑也都可以视为树状结构,因此采用树状结构的拓扑具有普遍意义.所采用拓扑图的结构如图 3 所示.图中与叶节点直接相连的链路带宽为 1 Mbps,与叶节点直接相连的链路传输时延为 10 ms,除与叶节点直接相连的链路外,内部链路带宽为 2 Mbps,内部链路的传输时延为 15 ms.内部节点采用 DropTail 的丢包策略,背景流量加入自相似流.在根节点向叶节点发送紧接分组对,紧接分组对内数据分组的时间间隔为 0.01 ms,分组对之间的时间间隔为 50 ms.

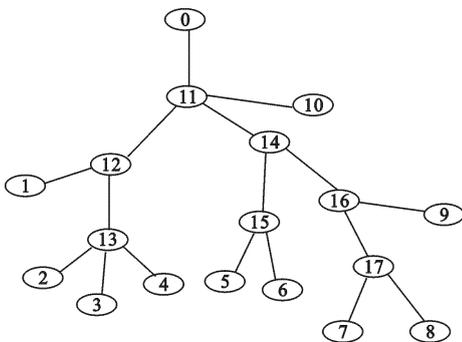


图 3 NS2 仿真拓扑

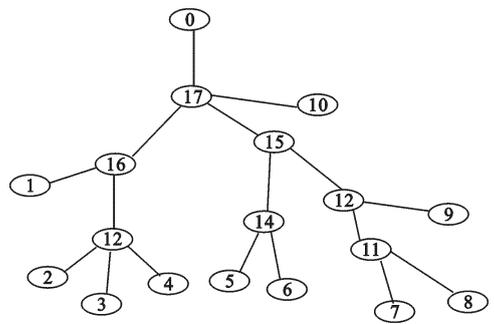


图 4 网络中载和重载时推断的拓扑

分别在网络负载适中时(丢包率大于等于 1% 小于 10%)和网络负载较重时(丢包率小于等于 10%)应用了拓扑推断算法.计算时延协方差和丢包率时发送紧接分组对的样本数为 25,对于每个叶节点对测量次数为 50 次.

在网络负载适中时,共进行了 50 次仿真实验,其中有 46 次得到图 4 的结果;网络负载较重时,共进行了 50 次仿真实验,其中有 43 次得到图 4 的结果,图中内部节点按推断的顺序进行编号.

表 1 列出了在同样网络负载的仿真环境下,分别以时延、丢包率和自适应综合参数作为节点相关性计算参数时正确推断网络拓扑的情况.

表 1 3 种参数的正确推断次数统计(50 次仿真)

	负载适中	负载较重	平均正确率/%		负载适中	负载较重	平均正确率/%
时延	46	35	81	综合参数	46	43	89
丢包率	21	39	62				

通过表 1 可以发现,采用自适应方法计算的相关性在推断网络拓扑时正确率要高于基于单一参数计算的相关性推断的正确率,例如在网络负载适中时基于综合参数推断的效果和基于时延推断的效果相当,但在负载较重时基于综合参数推断的正确率明显要高于基于时延推断的正确率,而综合参数与丢包率相比在网络负载适中和网络负载较重的情况下基于综合参数推断的正确率都比基于丢包率推断的正确率高.

以网络负载较重时 50 次仿真实验中的一次结果值作对比分析,图 5~图 7 分别为网络负载较重时基于时延、丢包率和综合参数的节点间的相关性值.采用了三维图形显示节点间的相关性值,把标准化后的时延协方差、成功

传输率和综合参数称为时延、丢包率和综合参数. 在三维图形中, 所有节点对以及节点对的相关性值都可在图形中清晰地显示, 在三维图形中可以根据  $X$  轴为叶节点  $i$ ,  $Y$  轴为叶节点  $j$ ,  $Z$  轴为节点对  $(i, j)$  的相关性值.

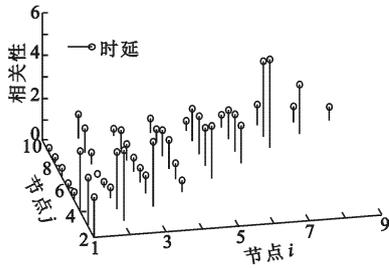


图 5 基于时延的相关性值

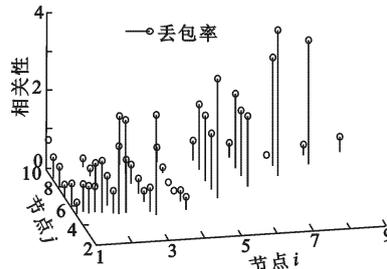


图 6 基于丢包率的相关性值

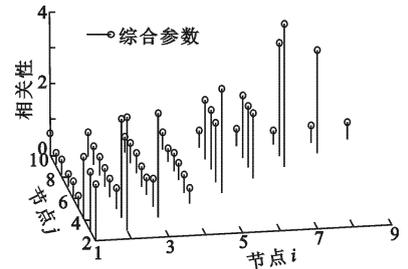


图 7 基于自适应计算的相关性值

对比图 5~图 7 可以发现, 在网络负载较重时, 采用丢包率计算的节点间相关性值比采用时延计算节点间相关性值时效果要好. 例如仿真拓扑结构中节点 2~4 共享链路相同, 但在图 5 中 3 个节点之间的相关性值差别较大; 在图 6 中, 节点 2~4 之间的根据丢包率计算的相关性差别小于基于时延计算的相关性差别, 而在图 7 中根据综合参数计算的共享链路相同的节点间的相关性差别非常小. 对比 3 种参数计算的节点间相关性发现, 基于综合参数计算的节点间相关性的值比基于时延和丢包率计算的节点间相关性值要准确.

通过多次的仿真实验表明, 采用自适应算法计算的节点相关性值在网络负载适中和网络负载较重时均能准确地反映节点共享链路的情况.

## 5 小 结

根据当前拓扑推断算法依据单一性能参数的缺点, 设计了自适应网络拓扑推断算法, 该算法实现简单, 并且无需增加测量流量. NS2 仿真实验表明在网络负载适中或网络负载较重时自适应算法都能较准确地推断出网络拓扑结构.

### 参考文献:

- [1] Coates M, Hero A, Nowak R, et al. Internet Tomography[J]. IEEE Signal Process Mag, 2002, 19(3): 47-65.
- [2] Coatro R, Coates M, Liang G, et al. Network Tomography: Recent Developments[J]. Statistical Science, 2004, 19(3): 499-517.
- [3] Coates M, Nowak R. Network Loss Inference Using Unicast End-to-end Measurement[C]//ITC Conf on IP Traffic, Modelling and Management. Monterey: Monterey: TTC Conf on Traffic, Modelling and Mangement, 2000: 1281-1289.
- [4] Meng-Fu S, Alfred O. Unicast-based Inference of Network Link Delay Distributions with Finite Mixture Models[J]. IEEE Trans on Signal Processing, 2003(51): 2219-2228.
- [5] Ratnasamy S, Mccanne S. Inference of Multicast Routing Trees and Bottleneck Bandwidths Using Eend-to-endmeasurements[C]//IEEE INFOCOM. New York: IEEE, 1999: 1353-1360.
- [6] Coates M, Castro R, Nowak R. Maximum Likelihood Network Topology Identification from Edge-based Unicast Measurements[C]//ACM Sigmetric. Marina Del Rey: ACM Sigmetric, 2002: 1015-1023.
- [7] Castro R, Coates M, Nowak R. Likelihood Based Hierarchical Clustering[J]. IEEE Trans on Signal Processing, 2004, 52(8): 2308-2321.
- [8] Meng-Fu S, Alfred O. Topology Discovery on Unicast Networks: a Hierarchical Approach Based on End-to-End Measurements[J/OL]. [2006-12-15]. <http://www.eecs.umich.edu/~msim/Publications/cspl-357.ps.pdf>.
- [9] Duffield N, Horowitz J, Presti F, et al. Multicast Topology Inference from End-to-end Measurements[C]//ITC Seminar on IP Traffic, Measurement and Modelling. Monterey: TTC Conf on Traffic, Modelling and Mangement, 2000: 2384-2394.
- [10] Nsnam Sourceforge Project. The Network Simulator 2 [EB/OL]. [2007-12-25]. <http://nsnam.isi.edu/nsnam/>.