

利用往返时延抖动的网络拓扑推断算法

赵洪华, 陈鸣

(解放军理工大学指挥自动化学院, 210007, 南京)

摘要: 为了克服基于端到端单向时延的拓扑推断需要时钟同步及节点间合作的限制, 提出了一种利用往返时延抖动的拓扑推断算法. 首先定义了四元组列车, 其由 4 个长度相同的 ping 分组组成, 4 个 ping 分组组成 2 个相邻的紧接分组对, 2 个紧接分组对的目标地址相同. 在空间独立性、时间独立性的条件下, 通过四元组列车测量获得的往返时延抖动可以计算节点间的相关性, 再根据节点间的相关性便可推断节点间共享链路, 从而推断出网络拓扑. 理论分析与仿真结果表明, 所提算法的收敛速度高于基于端到端单向时延推断拓扑法, 并且只需要一个测量节点.

关键词: 网络; 往返时延抖动; 拓扑推断

中图分类号: TP393 **文献标志码:** A **文章编号:** 0253-987X(2009)06-0028-05

A Topology Inference Algorithm Using Round Delay Variation

ZHAO Honghua, CHEN Ming

(Institute of Command Automation, PLA University of Science & Technology, Nanjing 210007, China)

Abstract: In order to reduce the limits of time synchronization and cooperation between nodes in topology inference based on one way delay, a topology inference algorithm based on round delay variation is proposed. A 4-tuple packets train is defined and is composed of 2 packets pairs. Each packets pair is composed of 2 ping packets. The correlations among nodes are then calculated by round delay variation measured with the 4-tuple packets train. Then the links of sharing between nodes and the topology are inferred. The feasibility and correctness of the topology inference based on round delay variation are analyzed theoretically. The algorithm is tested through simulations by NS2. The results show that the convergence of topology inference based on round delay variation is faster than the convergence of topology inference based on one way delay, and the topology inference based on round delay variation needs only one measurement node.

Keywords: network; round delay variation; topology inference

网络层析成像^[1]可以通过端到端的技术推断网络的内部特性, 例如通过端到端的时延或丢包率来推断内部链路的时延或丢包率^[2-3]. 网络拓扑推断是根据网络中节点性能特性的相关性来推断网络的拓扑结构, 是网络层析成像技术的最新应用之一. 研究表明, 网络中节点的共享链路越多, 节点的性能特性越相近, 即相关性越大^[4]. 1999 年 Ratnasamy 在网络丢包测量中观察到了节点相关性的特性, 并将基于丢包率所观察到的节点相关性应用于多播网络拓

扑结构的推断中^[5]. Duffield 等人严格证明了基于节点相关性推断多播网络拓扑结构的正确性, 并在推断中应用了其他的网络性能参数^[6].

在单播网络拓扑推断中需要采用特殊的测量方法, 以计算节点间的相关性, 当前比较常用的测量方法有 2 种: 紧接分组对测量方法^[7]和“三明治”分组列车测量方法^[8], 但这 2 种测量方法都受到较多的限制.

为了减少单播网络拓扑推断的限制, 本文提出了一种基于往返时延抖动^[9-10]的推断算法, 并设计

了往返时延抖动的测量方法(文中称为四元组列车测量方法),该方法不需要节点的时钟同步和节点间的合作。

1 往返时延抖动测量方法

在单播网络中为了测量节点间的往返时延抖动,提出了往返四元组列车测量方法,简称为四元组列车测量方法,其定义如下。

(1)四元组列车由4个ping分组组成,4个ping分组组成2个相邻的紧接分组对,这2个紧接分组对的目标地址相同,设为 (x_1-y_1, x_2-y_2) 。

(2)在四元组列车中,ping分组 x_1, x_2 具有相同的目标地址,ping分组 y_1, y_2 具有相同的目标地址,4个分组的长度相同。

测量时有2个假定条件:①空间独立性,即数据分组在不同的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。

采用四元组列车测量往返时延抖动的典型事例如图1所示。

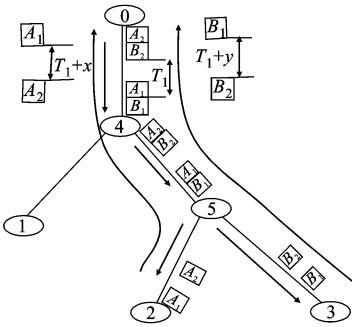


图1 四元组列车测量事例

设四元组列车经过的链路为 $L_1, L_2, \dots, L_n$, 各条链路的带宽为 $b_1, b_2, \dots, b_n$, 又设四元组列车中分组的长度为 $s(p)$, 则四元组列车内紧接分组对的时间间隔要满足

$$\frac{s(p)}{T_1} < \min(b_i) \quad 1 \leq i \leq n \quad (1)$$

测量时,设四元组列车为 (x_2-y_2, x_1-y_1) , 四元组列车内的2个紧接分组对分别为 x_1-y_1 和 x_2-y_2 , 紧接分组对之间的时间间隔为 T_1 , 紧接分组对内的时间间隔为0。四元组列车之间的时间间隔为 T , 测量源发送第1个四元组列车的时刻为 T_0 , 则第 N 个四元组列车中分组的发送时间为 $\{T_0, T_0+T, T_0+T_1, T_0+T_1+T\}+NT$ 。

四元组列车的目标地址为节点对地址 (D_x, D_y)

的,在源节点收到目标节点 D 的返回分组的时间为 $T_x = \{T_{x_1}, T_{x_2}, T_{2x_1}, T_{2x_2}, \dots, T_{ix_1}, T_{ix_2}, \dots, T_{Nx_1}, T_{Nx_2}\}$, 其中 T_{ix_1}, T_{ix_2} 表示第 i 个四元组列车中分组 x_1 和分组 x_2 的返回时间。

通过第 N 个四元组列车测量的到达 D_x 的往返时延抖动

$$T_{ix_N} = (T_{Nx_2} - T_{Nx_1}) - T_1 \quad (2)$$

可以计算到达目标 D_x 的所有往返时延抖动。根据式(1)和式(2)可知,计算时延抖动与分组的发送无关,因此时延抖动的测量不需要节点间保持时钟同步,也不需要节点间相互配合。用向量 T_{ix} 表示到达目标 D_x 的往返时延抖动。

同理,通过第 N 个四元组列车测量的到达 D_y 的往返时延抖动

$$T_{iy_N} = (T_{Ny_2} - T_{Ny_1}) - T_1 \quad (3)$$

可计算到达 D_y 的所有往返时延抖动。用向量 T_{iy} 表示到达 D_y 的所有往返时延抖动。

2 节点间相关性计算与拓扑推断

通过四元组列车测量的往返时延抖动可以计算节点间的相关性,如图2所示。设四元组列车的目标地址为节点对 (i, j) 的地址,令 $a(i, j)$ 表示节点 i 和节点 j 的父节点, Y_i, Y_j 分别表示测量源节点 s 到节点 i 、节点 j 的往返时延抖动向量, $Y_i^{i,j}, Y_j^{i,j}, \bar{Y}_i^{i,j}, \bar{Y}_j^{i,j}$ 分别表示测量源节点 s 到节点 i 和节点 j 、节点 i 和节点 j 到测量源节点 s 的单向时延抖动向量。又设 $X_a^{i,j}, \bar{X}_a^{i,j}, X_i^{i,j}, \bar{X}_i^{i,j}, X_j^{i,j}, \bar{X}_j^{i,j}$ 分别表示源节点 s 到节点 $a(i, j)$ 、节点 $a(i, j)$ 到源节点 s 、节点 $a(i, j)$ 到节点 i 、节点 i 到节点 $a(i, j)$ 、节点 $a(i, j)$ 到节点 j 、节点 j 到节点 $a(i, j)$ 的单向时延抖动向量。

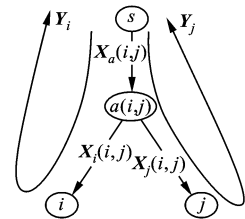


图2 根据时延抖动计算相关性

测量源节点 s 到节点 i 和节点 j 的往返时延抖动协方差,可以根据节点间的往返路径分为下面4种情况。

(1)源节点 s 到目标节点 i 的往返路径相同,源节点 s 到目标节点 j 的往返路径相同,不同链路分组相互独立,则有

$$\begin{aligned} \text{cov}(\mathbf{Y}_i, \mathbf{Y}_j) &= \text{cov}(\mathbf{X}_a^{i,j} + \mathbf{X}_i^{i,j} + \bar{\mathbf{X}}_i^{i,j} + \bar{\mathbf{X}}_a^{i,j}, \\ &\quad \mathbf{X}_a^{i,j} + \mathbf{X}_j^{i,j} + \bar{\mathbf{X}}_j^{i,j} + \bar{\mathbf{X}}_a^{i,j}) = \\ \text{cov}(\mathbf{X}_a^{i,j} + \mathbf{X}_i^{i,j})\text{cov}(\mathbf{X}_a^{i,j} + \mathbf{X}_j^{i,j} + \bar{\mathbf{X}}_j^{i,j} + \bar{\mathbf{X}}_a^{i,j}) + \\ \text{cov}(\bar{\mathbf{X}}_i^{i,j} + \bar{\mathbf{X}}_a^{i,j})\text{cov}(\mathbf{X}_a^{i,j} + \mathbf{X}_j^{i,j} + \bar{\mathbf{X}}_j^{i,j} + \bar{\mathbf{X}}_a^{i,j}) \Rightarrow \\ \text{cov}(\mathbf{Y}_i, \mathbf{Y}_j) &= \text{var}(\mathbf{X}_a^{i,j}) + \text{var}(\bar{\mathbf{X}}_a^{i,j}) \quad (4) \end{aligned}$$

(2)源节点 s 到目标节点 i 的往返路径不同,源节点 s 到目标节点 j 的往返路径相同,不同链路分组相互独立,则有

$$\begin{aligned} \text{cov}(\mathbf{Y}_i, \mathbf{Y}_j) &= \text{cov}(\mathbf{X}_a^{i,j} + \mathbf{X}_i^{i,j} + \bar{\mathbf{Y}}_i^{i,j}, \mathbf{X}_a^{i,j} + \\ &\quad \mathbf{X}_j^{i,j} + \bar{\mathbf{X}}_j^{i,j} + \bar{\mathbf{X}}_a^{i,j}) = \text{var}(\mathbf{X}_a^{i,j}) \quad (5) \end{aligned}$$

(3)源节点 s 到目标节点 i 的往返路径相同,源节点 s 到目标节点 j 的往返路径不同,不同链路分组相互独立,则有

$$\begin{aligned} \text{cov}(\mathbf{Y}_i, \mathbf{Y}_j) &= \text{cov}(\mathbf{X}_a^{i,j} + \mathbf{X}_i^{i,j} + \bar{\mathbf{X}}_i^{i,j} + \bar{\mathbf{X}}_a^{i,j}, \\ &\quad \mathbf{X}_a^{i,j} + \mathbf{X}_j^{i,j} + \bar{\mathbf{Y}}_j^{i,j}) = \text{var}(\mathbf{X}_a^{i,j}) \quad (6) \end{aligned}$$

(4)源节点 s 到目标节点 i 的往返路径不同,源节点 s 到目标节点 j 的往返路径不同,不同链路分组相互独立,则有

$$\begin{aligned} \text{cov}(\mathbf{Y}_i, \mathbf{Y}_j) &= \text{cov}(\mathbf{X}_a^{i,j} + \mathbf{X}_i^{i,j} + \bar{\mathbf{Y}}_i^{i,j}, \\ &\quad \mathbf{X}_a^{i,j} + \mathbf{X}_j^{i,j} + \bar{\mathbf{Y}}_j^{i,j}) = \text{var}(\mathbf{X}_a^{i,j}) \quad (7) \end{aligned}$$

式(5)~式(7)的证明同式(4)。

根据式(4)~式(7)可知:测量源节点到测量目标节点对的往返时延抖动协方差,是测量目标节点对共享链路的时延抖动方差之和;节点对的共享链路越多,往返时延抖动的协方差越大。往返时延抖动相对于时延更能反映网络中分组的统计特性。

设紧接分组对测量的目标节点对 (i, j) 的往返时延抖动协方差有 N 组样本,往返时延抖动协方差样本为 $(\hat{J}_1^{(i,j)}, \hat{J}_2^{(i,j)}, \dots, \hat{J}_N^{(i,j)})$, 节点对的相关性度量值为

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

拓扑推断算法是根据 $J_{i,j}$ 来推断节点间共享链路的情况,从而推断网络拓扑,拓扑推断算法如图3所示。

聚类时,以节点相关性最小值作为聚类阈值,拓扑推断算法采用聚类方式,计算复杂度为 $O(n)$ 。

3 NS2 仿真

为了验证往返时延抖动拓扑推断算法的健壮性和有效性,在 NS2^[11] 环境下进行了仿真实验。

仿真采用了树状结构的网络拓扑,如图4所示。由于基于单个发送节点、多个接收节点的拓扑结构

```

输入: 根节点{0}, 叶节点集  $V_r = \{1\}, \dots, \{M\}$ , 边集  $L = \emptyset$ , 节点相关性集合  $\bar{J} = (\bar{J}_{1,2}, \bar{J}_{1,3}, \dots, \bar{J}_{i,j}, \dots, \bar{J}_{m-1,m})$ 
输出:  $T = (V, L)$ 
While  $|V_r| > 1$  do
  Begin
    根据  $J$  对节点集划分  $V_r = A_1 \cup \dots \cup A_i \cup \dots \cup A_n$ , 对于每一个集合  $A_i$ 
     $\forall i \forall j (i \in A_i, j \in A_j) \rightarrow \bar{J}_{i,j} > \bar{J}_{i,m} (\forall m, m \notin A_i)$ ;
    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  $J$ , 减去  $J$  中  $A_i$  的值, 加入新值  $\bar{J}_{m,x}$ ,
             $\bar{J}_{m,x} (x \notin A_i) = \max(\bar{J}_{m,x} (x \notin A_i))$ ;
             $V = A_i \cup \{m\}$ ;
             $L = \{(m, x), x \in A_i\}$ ;
             $V_r = \{V_r - A_i\} \cup \{m\}$ ;
          End;
        End;
      End;
    End;
  End;
End;
```

图3 拓扑推断算法

均为树状结构,因此采用它具有普遍意义。与叶节点相连的链路带宽为 1 Mb/s, 链路时延为 10 ms; 内部链路带宽为 2 Mb/s, 链路时延为 15 ms。内部节点采用随机早期丢包 (RED) 的丢包策略, 背景流量为自相似流和泊松流。

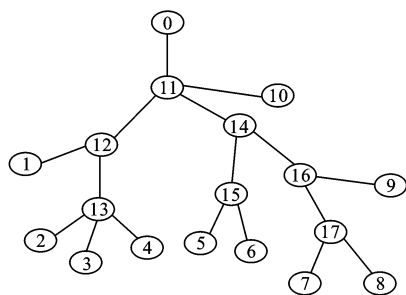


图4 NS2 仿真拓扑结构图

在仿真实验中,对于每一个叶节点对,每组发送 20 个四元组列车,测量的组数从 1 增加到 25。

分别在网络负载较小 (丢包率 $< 1\%$)、负载适中 ($1\% \leq \text{丢包率} \leq 10\%$) 和负载较重 (丢包率 $> 10\%$) 的情况下进行了多次仿真实验。

采用树编辑距离^[12]来评价推断拓扑树与原拓扑树的差别,树编辑距离 (步骤数) 就是 T_1 转换为 T_2 所需要的最小花费。

为了便于比较,将根据往返时延抖动计算的节点间相关性称为 RDV, 根据单向时延计算的节点间相关性称为 OWD。图5~图7为仿真结果。

当网络负载较小或适中时,采用往返时延抖动

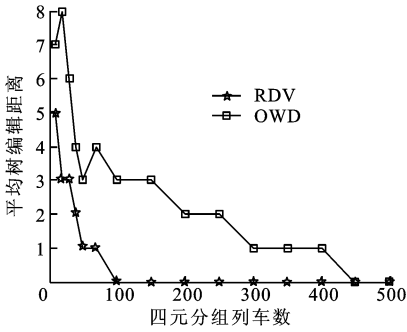


图 5 网络负载较小时树编辑距离收敛情况

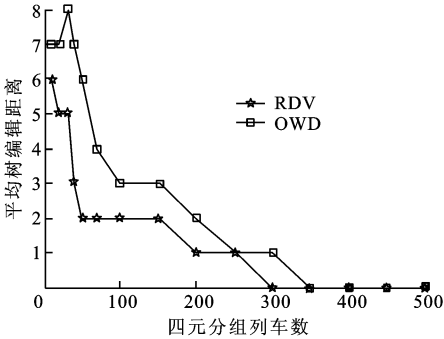


图 6 网络负载适中时树编辑距离收敛情况

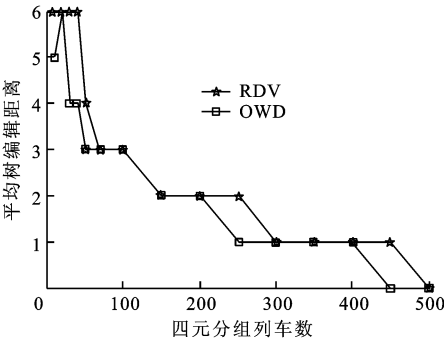


图 7 网络负载较大时树编辑距离收敛情况

推断拓扑的效果优于采用单向时延推断拓扑的效果,并且往返时延抖动的测量不需要节点的时钟同步和节点间的合作. 当网络负载较大时,采用往返时延抖动作为参数推断拓扑的树编辑距离收敛速度不如采用单向时延推断拓扑的情况.

通过不同网络负载下的仿真实验可知,往返时延抖动的测量不需要节点的时钟同步,并且基于往返时延抖动推断拓扑结构的收敛速度优于单向时延推断拓扑的情况.

4 结束语

采用往返时延抖动推断网络拓扑比采用单向时

延推断网络拓扑受到的限制更少,且不需要节点间的时钟同步和节点间的合作. 在网络负载较小或适中的环境下,用往返时延抖动推断网络拓扑的收敛速度更快.

参考文献:

[1] CASTRO R, COATES M, LIANG G, et al. Network tomography: recent developments[J]. Statistical Science, 2004, 52(3): 499-517.

[2] LAWRENCE E, MICHAELIDIS G, NAIR V. Network delay tomography using flexi cast experiments [J]. Journal of the Royal Statistical Society Series: B, 2006, 68:785-813.

[3] DUFFIELD N G, LO F, PAXSON V, et al. Network loss tomography using striped unicast probes [J]. IEEE/ACM Transactions on Networking, 2006, 14 (4): 697-710.

[4] RATNASAMY S, MCCANNE S. Inference of multi-cast routing trees and bottleneck bandwidths using end-to-end measurements[C] // Proceedings of IEEE INFOCOM 1999. Piscataway, NJ, USA: IEEE, 1999: 353-360.

[5] DUFFIELD N, HOROWITZ J, PRESTI L F, et al. Multicast topology inference from measured end-to-end loss [J]. IEEE Transactions on Information Theory, 2002, 48(1): 26-45.

[6] DUFFIELD N, HOROWITZ J, PRESTI L F, et al. Multicast topology inference from end-to-end measurements[C]//The 13th ITC Seminar on IP Traffic Measurement, Modelling and Management, Würzburg, Germany: University of Würzburg, 2000: 654-674.

[7] SHIH M, HERO A O. Hierarchical inference of unicast network topologies based on end-to-end measurements [J]. IEEE Transactions on Signal Processing, 2007, 55(5): 1708-1718.

[8] COATES M, CASTRO R, NOWAK R. Maximum likelihood network topology identification from edge-based unicast measurements[J]. SIGMETRICS Perform Eval Rev, 2002, 30(1): 11-20.

[9] DEMICHELIS C, CHIMENTO P. IP packet delay variation metric for IPPM [EB/OL]. [2008-01-10]. <http://www.ietf.org/internetdrafts/draft-ietf-ippm-ipdv-09.txt>.

[10] 王立,李增智,宋承谦,等. 一种满足多服务质量约束的动态组播路由算法[J]. 西安交通大学学报, 2004, 38 (4):420-423.

WANG Li, LI Zengzhi, SONG Chengqian, et al. Dy-

- dynamic multicast routing algorithm satisfying multiple QoS constraints [J]. Journal of Xi'an Jiaotong University, 2004, 38(4): 420-423.
- [11] Nsnam Sourceforge Project. The network simulator 2 [EB/OL]. [2008-01-10]. <http://nsnam.isi.edu/nsnam/>.
- [12] PHILIP B. A survey on tree edit distance and related problems [EB/OL]. [2008-01-10]. <http://grfia.dlsi.ua.es/ml/algorithms/references/editsurvey-bille.pdf>.

(编辑 苗凌)