

基于本地域信息的时延约束 Steiner 树算法

来卫国, 李 鸥

(解放军信息工程大学信息工程学院通信工程系, 郑州 450002)

摘 要: 提出一种基于本地域 QoS 信息的时延约束 Steiner 树算法, 通过最短代价路径构造组播树, 源节点根据目的节点及源区域边界路由提供的最短时延路径信息, 对剩余目的进行集中式路由计算, 并由控制消息完成组播树的配置及环路删除。该算法不依赖于全局性的 QoS 信息, 从而避免大量控制开销与陈旧 QoS 信息的影响。仿真实验结果表明, 该算法具有较高的加入成功率以及较合理的控制开销。

关键词: 时延约束; Steiner 树; QoS 组播; 源区域边界路由器

Algorithm for Delay-constrained Steiner Tree Based on Local Domain Information

LAI Wei-guo, LI Ou

(Dept. of Communication Engineering, College of Information Engineering, PLA Information Engineering University, Zhengzhou 450002)

【Abstract】 A novel delay-constrained Steiner tree algorithm based on local domain QoS information is presented, which constructs a multicast tree along the shortest cost paths, and on the basis of the least delay path information provided by group members and Source Area Border Routers(SABR), the source node computes the best routes for remain members. Then, the control messages are sent out to configure the multicast tree and break loops. This algorithm is not dependent on global QoS information, so it can cut down control overheads greatly and be immune from stale QoS information. Simulation experimental results show this algorithm has high join success rate and reasonable control overheads.

【Key words】 delay-constrained; Steiner tree; QoS multicast; Source Area Border Router(SABR)

1 概述

随着视频会议、视频点播以及在线游戏等实时组播业务的发展, 时延约束 Steiner 树问题成为组播研究的热点问题之一^[1-6]。该问题是求解满足端到端的时延要求且代价最小的组播树, 根据节点掌握 QoS 信息的种类不同, 时延约束 Steiner 树算法大体可分为 3 类:

(1) 节点仅知道至其他节点的最短代价路径(LCP)及其代价值, 如 QMRP 算法^[1]、SoMR 算法^[2], 由于该类算法主要通过向目的节点发送探测分组来找路, 因此具有一定的盲目性与不确定性。

(2) 节点知道 LCP 路径及其代价与时延值, 如 Sriram's 算法^[3]、Jia's 算法^[4], 由于该类算法在找路过程中仅使用 LCP 路径信息, 因此效果并不理想。

(3) 节点不但知道 LCP 路径信息, 且假设链路的时延信息精确可知, 从而知道最短时延路径(LDP)信息, 如 Ural's 算法^[5]、Feng's 算法^[6], 该类算法精度最高。

链路时延信息可以利用 OSPF 等协议的可靠洪泛获得, 然而频繁洪泛占用过多的网络带宽资源, 导致网络性能下降。因此, 必须在时延信息的精确性与控制开销之间折中^[7]。

本文提出基于本地域 QoS 信息计算的时延约束 Steiner 树算法。该算法仅在局部域进行时延信息的洪泛, 源节点通过控制消息获得相关信息并进行全局 QoS 路由计算。由于洪泛开销正比于网络规模的平方, 因此局部洪泛可以在合理的开销下获得足够精确的时延信息。

2 时延约束 Steiner 树问题

设路由网络用无向图 $G(V, E)$ 表示, 其中, V 为顶点集合;

E 为边集合; $c(e)$, $d(e)$ 分别表示边 e 的代价与时延值。对于组播树 T , $P(s, u)$ 表示沿组播树 T 从源节点 s 到 u 的路径, $c(T) = \sum_{l \in T} c(l)$ 为 T 的代价。

定义(时延约束 Steiner 树问题) 给定组播源节点 s 、目的节点集合 D 及时延约束值 Δ , 求一棵覆盖 s 与 D 的组播树, 该树满足 $\sum_{l \in P(s, u)} d(l) \leq \Delta, \forall u \in D$, 且对任意一棵满足上述条件的组播树 T' , 均有 $c(T) \leq c(T')$ 。

3 本文算法

3.1 基本思想

将整个路由域划分为若干区域, 其中, 骨干区域位于路由域的中心, 本地域与骨干区域边界上的路由器称为区域边界路由器(ABR)。由于链路代价信息较稳定, 因此使用较低的频率在整个路由域内洪泛, 而链路时延信息以较高的频率在本地域内洪泛。利用上述链路信息, 每个节点可以构造以代价为度量的全局路由表及以时延为度量的区域路由表, 而 ABR 具有至少 2 个区域路由表。

本算法分为 3 个阶段:

- (1) 采用并行方式, 沿 LCP 路径构造组播树;
- (2) 源节点计算到剩余目的节点的时延约束最小代价路径;

基金项目: 中国下一代互联网(CNGI)示范工程基金资助项目(CNGI-04-10-1D)

作者简介: 来卫国(1972 -), 男, 博士研究生, 主研方向: 因特网组播技术, 网络优化技术; 李 鸥, 博士生导师

收稿日期: 2008-09-15 **E-mail:** laiwg@163.com

(3)沿上述路径构造组播树,并进行去圈处理。同时使用指定源组播模式,以 (s, G) 表示一个组播通道。

3.2 算法第1阶段

源节点收到来自应用层的组播请求后,生成一个 SETUP 消息,该消息包含组播地址 G 、组播源地址 s 、时延要求 Δ 、目的节点列表。源节点启动计时器并处理该消息,节点在收到 SETUP 消息后,进行如下处理:

- (1)更新消息的累积时延值;
- (2)查询全局路由表,将目的节点列表按照 LCP 路径的下一跳划分成簇,为每簇生成一个包含本簇成员的 SETUP 消息,并发送至簇首节点;
- (3)建立组播转发表(MFT)表项,并正确设置表项累积时延值、上游接口及转发接口列表。

目的节点收到 SETUP 消息后,若发现消息的累积时延值大于时延要求 Δ (称该节点为 RN 节点),则:

- (1)生成 ACK 消息,并令消息的删除标志为真;
- (2)查询区域路由表,将到本区域所有 ABR(即 DABR)的 LDP 路径信息添加到消息中,并发送至源节点。

沿途节点收到 ACK 消息后,进行如下处理:

- (1)查询得到该组播通道的 MFT 表项;
- (2)如消息删除标志为真,则从转发接口列表中删除消息的进入接口;
- (3)如转发接口为空,则删除 MFT 表项;否则令消息的删除标志为否;
- (4)经 MFT 表项上游接口转发该消息。

源节点收到 ACK 消息后,将相关信息存入 ABR 表。待计时器到后,如未收到 ACK 消息,表示所有目的均成功加入,算法结束;否则进入算法的第 2 阶段。

3.3 算法第2阶段

源节点向所有本区域 ABR(即 SABR)发送包含 ABR 表的 QUERY 消息,后者查询骨干区区域路由表,得到至表中各 DABR 的 LDP 路径信息,并用 QUERY-ACK 消息回复源节点。源节点按如下代码计算到节点 w 的时延约束最小代价路径:

```
w.COST=MAX;
for each ser in {SABR}
    for each der in {DABR}
        DELAY= D(Pld(s, ser))+D(Pld(ser, der))+D(Pld(der, dest));
        COST= C(Pld(s, ser))+C(Pld(ser, der))+C(Pld(der, dest));
        if (DELAY ≤ Δ and COST < w.COST) then
            w.COST=COST; w.SABR=ser; w.DABR=der;
```

其中, {SABR}, {DABR} 分别为 SABR 集合与 DABR 集合; $P_{ld}(u, v)$ 为 $u \sim v$ 的最小时延路径; $C(x)$, $D(x)$ 分别表示路径 x 的代价与时延。

最终,源节点得到类似表 1 所示的 RN 节点信息表。

表 1 RN 节点信息表

RN 节点	SABR	DABR
A	ABR1	ABR6
B	ABR2	ABR4
C	ABR2	ABR3
D	ABR2	ABR3

3.4 算法第3阶段

源节点将 RN 节点信息表添加到 RNSETUP 消息中,并发送该消息。节点对该消息处理步骤略。

图 1 为本算法第 3 阶段的示意图,而 RN 节点信息表如表 1 所示。源节点分别向 SABR1 和 SABR2 发送 RNSETUP

消息,给前者的消息包含节点 A,给后者的消息包含节点 B, C, D。ABR1 将收到的消息直接发送至 ABR6,由后者发送至节点 A。ABR2 将包含节点 B 的 RNSETUP 消息发送给 ABR4,将包含节点 C, D 的 RNSETUP 消息发送至 ABR3。最后, ABR3, ABR4 分别将 RNSETUP 消息发送至各 RN 节点。

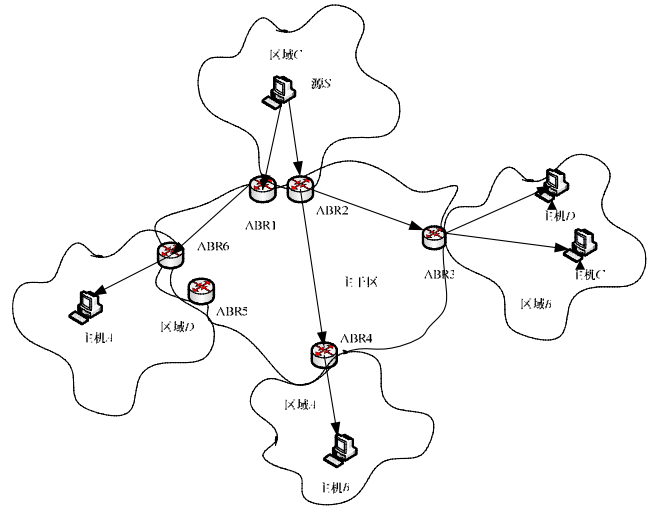


图 1 本文算法第 3 阶段示意图

4 算法分析

定理 1 假设本地区域间没有直接相连,若满足时延约束的组播树存在,则本算法一定得到可行解。

证明 如果源节点 s , 目的节点 w 均在同一本地域内,则 $s \sim w$ 的 LDP 路径必然为可行解。考虑 s, w 在不同区域内,假设存在一条可行路径 $P(s, w)$, 而本算法失败。由于各个本地区域间不存在直接的链路连接,因此该路径必然经过一个 SABR 与一个 DABR, 分别令其为 SER_i 与 DER_j , 即 $P(s, w) = P(s, SER_i) + P(SER_i, DER_j) + P(DER_j, w)$, 由最小时延路径的性质可知:

$$D(P_{ld}(s, SER_i)) + D(P_{ld}(SER_i, DER_j)) + D(P_{ld}(DER_j, w)) < D(P(s, w)) \quad (1)$$

另外,本算法失败意味着从 s 经任意 SABR 与 DABR 组合到达 w 的路径均不满足时延要求,显然有:

$$D(P_{ld}(s, SER_i)) + D(P_{ld}(SER_i, DER_j)) + D(P_{ld}(DER_j, w)) > \Delta \quad (2)$$

由于式(1)与式(2)矛盾,因此定理 1 得证。

定理 2 假设网络节点总数为 N , 目的数为 k , 则本算法的控制报文开销(控制报文总跳数)为 $O(3kn)$ 。

证明 从源节点到任意目的节点的 LCP(或者 LDP)路径最大跳数为 $N-1$ 。在最坏情况下,所有目的节点均是在算法第 2 阶段找到可行路径。此时, SETUP 消息、ACK 消息及 RNSETUP 消息的最大数均为 k , 最大开销均为 $k(N-1)$, 所以,本算法的总控制开销为 $O(3kn)$ 。

由于 SETUP 消息及 SETUP2 消息均以最大共享路径方式发送,因此本算法的实际开销远小于 $O(3kn)$ 。

5 仿真实验与分析

使用 NS2 软件仿真 SPT 算法、Jia's 算法及 LDDCM 算法,仿真场景由 1 个骨干区和 4 个本地区域组成。各个区域的拓扑用 GT-ITM 软件产生,2 个节点间存在边的概率为 $P(u, v) = \alpha e^{-d/(\beta L)}$, 其中, d 为两节点间距离; $L = \sqrt{2}Scale$ 为节点间最大距离; $Scale$ 为区域节点数。主要参数如表 2 所示,其中, B 为链路带宽; N 为本地域节点数。仿真结果均为

30次重复实验的平均值。

表2 仿真参数列表

骨干区参数	非骨干区参数	链路传播时延/ms	链路代价
$\alpha=0.7, \beta=0.5,$ $B=10 \text{ Mb}, N=20$	$\alpha=0.7, \beta=0.5,$ $B=5 \text{ Mb}, N=30$	$10 \times \text{链路长度}$	1

首先研究不同时延约束下各算法的性能，每次随机选取15个目的节点，时延约束 Δ 从0.1s递增至1.2s。

由图2可见，本文算法的加入成功率明显高于其他算法，这是因为Jia's算法与SPT算法仅考虑LCP路径，而本文算法同时考虑LCP路径与LDP路径。

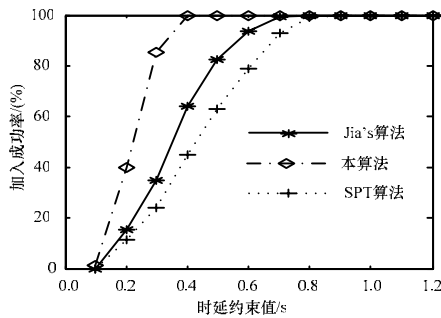


图2 时延约束与加入成功率的关系

图3为各算法的控制报文开销。本算法的控制开销在 $\Delta=0.3$ 时达到出现峰值，且在此之前，随 Δ 的增大以及源节点计算出可行路径的节点数增加，RNS SETUP消息数量也相应增加。当 $\Delta > 0.3$ 并继续增大时，在算法第1阶段成功加入的目的数增加，控制消息数量减少。在 $\Delta=0.9$ s后，所有目的节点均在第1阶段成功加入，本算法开销与SPT算法一样。

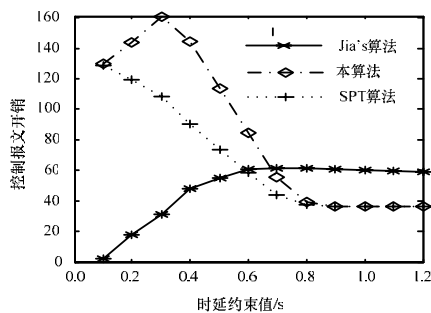


图3 时延约束与控制报文开销的关系

图4、图5显示了不同节点数量下各算法的性能。仿真参数为 $\Delta=0.25$ ，目的节点数从5~40以步进5递增。

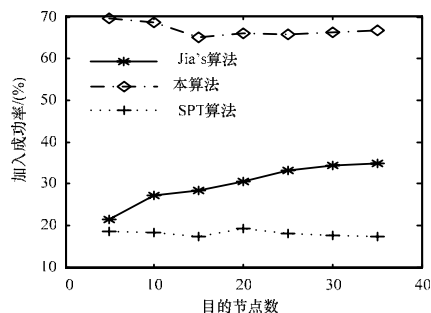


图4 目的节点数与加入成功率的关系

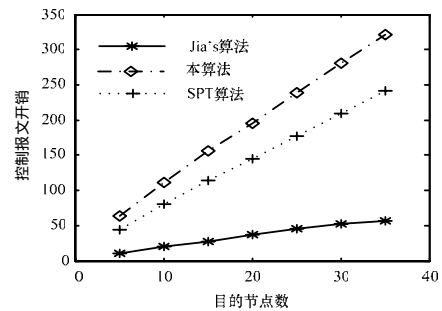


图5 目的节点数与控制报文开销的关系

由图4可见，本文算法与SPT算法的加入成功率仅在较小的范围内波动，且前者成功率较高。对于这2个算法，一个目的节点加入是否成功仅取决于该节点的物理位置及网络拓扑。而Jia's算法的成功率随着目的节点数的增加而增加，这是因为随着组播树上节点数量的增多，组播树到任意目的节点的可行路径数量增多。

图5显示出3种算法的控制开销均与目的节点数量呈线性关系，而本算法的斜率最大，因为它通过控制消息来获取全局信息，所以控制开销必然要大。虽然仿真中Jia's算法的控制开销较小，但是其需要进行全局QoS信息洪泛，实际开销要大很多。

6 结束语

本文提出基于本地区域QoS信息的时延约束Steiner树算法。该算法通过控制信息的交互获得全局性知识，从而可以在苛刻的约束条件下获得较高的加入成功率。

参考文献

- [1] Chen Shigang, Shavitt Y. A QoS-aware Multicast Routing Protocol[C]/Proc. of IEEE INFOCOM'00. [S. l.]: IEEE Press, 2000.
- [2] Chen Shigang, Shavitt Y. A Scalable Distributed QoS Multicast Routing Protocol[C]/Proc. of IEEE Int'l Conf. on Communications. Paris, France: [s. n.], 2004.
- [3] Jia Xiaohua. A Distributed Algorithm of Delay-bounded Multicast Routing for Multimedia Applications in Wide Area Networks[J]. IEEE/ACM Trans. on Networking, 1998, 6(6): 828-837.
- [4] Sriram R, Manimaran G. Algorithms for Delay-constrained Low-cost Multicast Tree Construction[J]. Computer Communications, 1998, 21(18): 1693-1706.
- [5] Ural H, Zhu Keqin. An Efficient Distributed QoS-based Multicast Routing Algorithm[C]/Proc. of Performance, Computing, and Communications Conference. Phoenix, Arizona, USA: [s. n.], 2002.
- [6] Feng Gang. Efficient Multicast Routing with Delay Constraints[J]. International Journal of Communication Systems, 1999, 12(2): 181-195.
- [7] Srihari N, Srivatsan V, Zhang Zhili. On Localized Control in QoS Routing[J]. IEEE Trans. on Automatic Control, 2002, 47(6): 1027-1032.

编辑 陈文