

## 双邻域查找组播聚合算法\*

汪学舜, 余少华, 戴锦友

(武汉邮电科学研究院, 湖北 武汉 430074)

通讯作者: 汪学舜, E-mail: wang\_xueshun@163.com, http://www.wri.com.cn

**摘要:** 光传输网络中聚合组播问题是一个完全 NP 难问题, 提出了一种解决聚合组播问题的双邻域查找算法. 该算法使得生成的聚合树数量在满足波长约束的前提下, 带宽浪费比率尽可能地小. 基于贪婪策略定义了一种优先聚合规则以生成初始解; 定义了两种邻域结构, 使邻域查找具有效率; 提出了跳坑策略以跳出局部最优解并且将查找引向有希望的方向. 模拟实验结果表明: 该算法可以有效地进行组播树的聚合, 当轻载时, 组播阻塞比率始终为 0; 当重载时, 与其他算法相比, 平均带宽浪费比率降低 25% 以上. 因此, 对不同的网络状况都能获得较好的性能.

**关键词:** 全光传输网络; 组播聚合; 拟人跳坑策略; 双邻域查找; 波长带宽分配

中图法分类号: TP393 文献标识码: A

中文引用格式: 汪学舜, 余少华, 戴锦友. 双邻域查找组播聚合算法. 软件学报, 2013, 24(2): 243-254. <http://www.jos.org.cn/1000-9825/4247.htm>

英文引用格式: Wang XS, Yu SH, Dai JY. Double neighborhood search algorithm in multicast aggregation. Ruanjian Xuebao/Journal of Software, 2013, 24(2): 243-254 (in Chinese). <http://www.jos.org.cn/1000-9825/4247.htm>

### Double Neighborhood Search Algorithm in Multicast Aggregation

WANG Xue-Shun, YU Shao-Hua, DAI Jin-You

(Wuhan Research Institute of Post and Telecommunication, Wuhan 430074, China)

Corresponding author: WANG Xue-Shun, E-mail: wang\_xueshun@163.com, http://www.wri.com.cn

**Abstract:** The problem of aggregated multicast in optical transmission networks has been known to be a complete NP-hard problem. A double neighborhood search algorithm (DNSA) for solving the multicast aggregation problem is presented. The objective of this algorithm is to minimize the bandwidth wastage ratio that is subject to the constraints that the number of multicast aggregation tree is affected by the amount of wavelength. In this paper, a priority aggregate rule based on the greedy strategy is proposed to generate initial solution: two kind neighborhood structures are proposed to search effectively, and some off-trap strategies are proposed to jump from local optimal solution and carry the search to the feasible areas in promising directions. Simulation experiments show the double neighborhood search algorithm can aggregate multicast trees effectively. The multicast group blocking ratio is 0 at light load. Compared with other algorithms at heavy load, the average bandwidth wastage ratio has decreased more than 25%. The results indicate that improvements may be obtained in different network condition.

**Key words:** all-optical transmission network; multicast aggregation; quasi-human off-trap strategy; double neighborhood search; wavelength and bandwidth allocation

流媒体是将来网络的主要应用, 光纤传输是下一代网络中的主要组成. 流媒体一般通过组播方式传输, 但是若将单个组播流直接应用到光纤传输网的一个波长进行传输则效率较低<sup>[1]</sup>. 这是因为单个波长的传输能力远远超过单个组播流的带宽需求. 组播并没有得到广泛的部署, 可能是因为组播在可管理性、可靠性、安全性以及 QoS 控制等方面存在技术上的问题<sup>[2]</sup>.

\* 基金项目: 光纤通信技术和网络国家重点实验室开放基金(2010OCTN-03)

收稿时间: 2011-01-23; 定稿时间: 2012-04-16

由于组播主要根据组进行传输,每一个组播组都使用单棵分发树.需要管理组播组、资源(如维护转发状态需要的内存)和管理开销(如建立和维护组播树),如果网络中组播组数量巨大,则资源和管理开销都很大,网络性能会急剧下降,这个问题称为组播状态规模问题.随着多用户应用的增加,这种现象进一步恶化.

组播状态的大规模问题引起了研究者的注意:某些机制通过输入节点的无分支减少转发状态<sup>[3]</sup>,另有机机制通过转发状态聚合来减少转发状态的维护<sup>[4]</sup>.但这些机制只考虑了大规模状态的资源管理方面.最近,研究者们提出了聚合组播方法,同时对资源和控制进行管理<sup>[5,6]</sup>.这种机制下,多个组播聚合到一棵聚合树上,可以显著减少网络中树的数量和转发状态.

如何将多个组播聚合到一棵组播树上,是一个完全 NP 难问题<sup>[7]</sup>,目前的研究主要有 3 种方法.一种方法是聚合组播模型(AM)<sup>[8]</sup>,强制将多个组播组聚合到单棵树,从而达到减少组播状态的数量.这种方法对多波长传输并不适用,且带宽浪费较大.另一种方法是 WDM 网络中的组播疏导模型(MG),假定各种组播组需要不同带宽需求,尽力满足各种组播组,同时使网络开销最小<sup>[9]</sup>,但要求整个网络交换结构全部为疏导的结构,且只能用于单数据源.第 3 种方法是采用三叉图模型和整形线性规划(ILP)方法解决聚合问题<sup>[10]</sup>,但其应用的困难在于需要大量的资源,即使对小规模网络,解决聚合组播问题也是一种挑战.文献[11]通过启发式算法解决聚合组播的整形线性规划问题,降低了复杂度,但要求不同网络规模采用不同的算法,实现起来比较困难.

对于聚合组播问题,另有一些学者提出了其他有效的组播路由和波长带宽分配算法,文献[12]中通过扩展多协议标签交换解决网络中点到多点的动态连接问题,但该方法对聚合组播的方法并没有改进.文献[13]中试图通过查找网络中源到目的的一组链路和波长来建立源到目的的路由.文献[14]提出了通过不同分配组播场景合适的波长使总代价最小.这两种方法主要适用于静态网络,不能适应网络的动态变化.

对光传输的组播聚合问题,可定义如下:给定系列头终端(即组播源),可提供一系列内容(即组播流)并需要相应的传输带宽,一系列请求接收内容的接入路由器(即组播目的),查找一系列聚合树满足最优的组播流聚合传输<sup>[15]</sup>,并能将聚合树映射到相应的波长.

本文中,目标是找到适合波长和带宽分配的优化组播聚合树.有以下假定:

- 1) 聚合树都是单向一对多的;
- 2) 一个组播组不能分成两棵或更多聚合树;
- 3) 每条链路都是双向的,每一个节点均支持所有波长的传输;
- 4) 所有组播组至少有 1 个接收者和 1 个数据源.

## 1 组播聚合模型

在光纤传输网络中,对波长和带宽分配,一般采取聚合组播的方法.本节首先给出包含头和尾终端的网络模型;然后对组播聚合问题进行描述,并说明组播聚合问题是一个 NP 难问题.

### 1.1 网络模型

图 1 说明了网络中,头终端的组播数据源将多媒体数据(如 IPTV 节目)编码为组播数据流,然后通过边界路由器发送到网络中(图 1 中的路由器 X).假定数据源与边界路由器之间的带宽是足够大的.头终端每一个组播流用字母  $y$  表示, $y$  后面括号中的数字表示该组播流需要的带宽,一般小于波长带宽.接入网的交换设备收集终端用户的需求,并将这些需求转发给组播数据源头终端,当接入交换设备从组播树收到组播流时,将该组播数据流转发给终端用户(图 1 中的路由器 Z),接入层交换设备实现对加入者的数据复制,对未加入者不转发.

网络中的组播能力可形式化为边带权重的无向图  $G(V, E, W)$ ,其中,  $V$  表示网络节点集合,  $E$  表示网络节点之间对可用波长的连接关系的边集合,  $W$  为链路集合  $E$  的代价.下文为了简化,将头终端的交换设备简化为头,尾终端的交换设备简化为尾,头作为聚合树的根,尾作为聚合树的叶子.图 1 中,尾终端用户请求组播数据是:  $Z_1$  请求组播流  $y_1$  和  $y_4$ ,  $Z_2$  请求组播流  $y_2$  和  $y_4$ ,  $Z_3$  请求组播  $y_1, y_3, y_4$  和  $y_5$ , 头终端  $X_1$  接入的组播源为  $y_1, y_2, y_3$  和  $y_4$ , 头终端  $X_2$  接入的组播源为  $y_2, y_3$  和  $y_5$ .如果仅仅使用 IP 组播机制,每一个组播流需要建立一棵组播树,则共需建立 5 棵组播树.但通过聚合之后,则只需要 3 棵聚合树,树 1 包含路由器 ACDE,根为 A,传播数据流  $y_1$  和  $y_4$ ,树 2 包含路由

器  $BFD$ , 根为  $B$ , 传播数据流  $y_2$ , 树 3 包含路由器  $BE$ , 根为  $B$ , 传播数据流  $y_3$  和  $y_5$ , 因此可以显著减少组播树的数量.

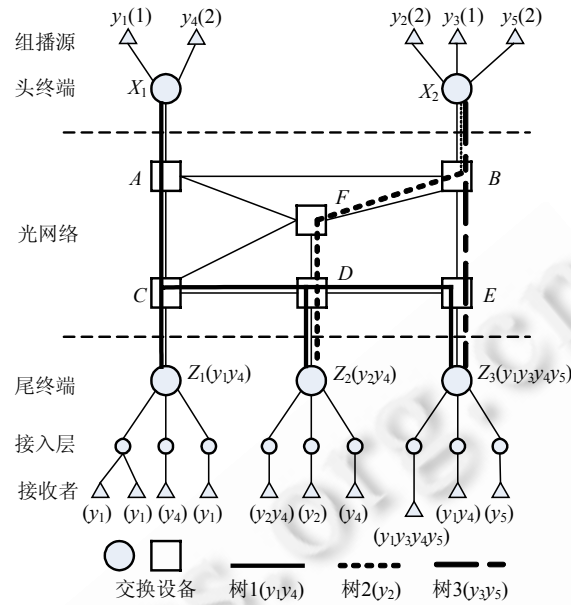


Fig.1 Network model of the multicast aggregation

图 1 组播聚合网络模型

下面说明一些术语和定义.

**树代价.** 给网络中每一条边  $(i,j)$  分配一个正的代价值  $c_{ij}=c_{ji}$ , 表示从节点  $i$  到节点  $j$  的传输代价. 对于由多条边组成的组播树  $t$ , 其总代价为

$$C(t) = \sum c_{ij} \quad (1)$$

如果每一条链路的代价均为 1, 则树的代价可简化为  $C(t)=|t|-1$ , 其中,  $|t|$  表示树  $t$  中的节点数.

**带宽浪费.** 对网络  $G(V,E,W)$ , 采用组播路由算法生成组播树, 如最短路径优先, 对组播组  $g$ , 设  $t^n(g)$  为由路由算法计算的组播树, 如果该组可由聚合树  $t(g)$  覆盖, 则其带宽浪费比率可按下式计算:

$$\delta(t, g) = \frac{C(t(g)) - C(t^n(g))}{C(t^n(g))} \quad (2)$$

其值直接反映了使用树  $t(g)$  代替原始树  $t^n(g)$  转发组  $g$  的数据时, 其带宽浪费的比率. 如果数据传输数量为  $D(g)$ , 则带宽浪费可计算为  $D(g) \times \delta(t, g) \times C(t^n(g))$ . 假定链路代价为 1, 则带宽浪费比率可按下式计算:

$$\delta(t, g) = \frac{|t(g)| - |t^n(g)|}{|t^n(g)| - 1} \quad (3)$$

为了减少带宽浪费, 一个组允许映射到一棵树的条件必须满足:  $\delta(t, g) \leq b_{th}$ , 其中,  $b_{th}$  为预定义的带宽浪费门限. 注意,  $t^n(g)$  不必是最小代价树. 实际上, 很多组播路由算法使用最短路径树, 如 MOSPF, DVMRP/PIM-DM, PIM-SSM 等, 也有些组播路由算法采用核心树 (如 CBT, PIM-SM 等). 聚合树  $t(g)$  可能比  $t^n(g)$  更为有效, 因此,  $\delta(t, g)$  可能为负值.

**组播组阻塞率.** 光纤传输网络中, 组播组数量达到一定程度后, 由于波长数量和波长带宽能力的限制, 部分组播组无法进行聚合, 不能聚合的组播组数据不能转发, 会在头终端丢弃. 设网络中丢弃的组播组数量为  $N_d$ , 组播组阻塞率定义如下:

$$\eta = N_d / N_g \quad (4)$$

## 1.2 问题描述

网络  $G(V, E, W)$  中, 所有组播组集合为  $g$ , 使用组播路由算法, 带宽浪费门限为  $b_{th}$ , 目标是找到  $N$  棵树  $t$ , 即  $N$  棵聚合树, 每一棵树覆盖节点集合不同, 从组  $g$  到树  $t(g)$  进行映射, 以便每一个组的节点集合能够被  $N$  棵树  $t$  覆盖, 并同时保证每一棵树带宽浪费小于门限.

光纤传输网络中, 如果希望建立一棵新的聚合树, 属于树的每一条边需要一个波长, 因此, 网络中每一条边  $(u, v)$  中可建立树的数目不超过全部可用波长数量  $w_{uv}$ , 称为波长约束. 形式化表示如下: 设  $T = \{t_1, t_2, \dots, t_k, \dots\}$ ,  $t_k$  为网络中建立的聚合树, 定义二进制变量:

$$\delta_{uvk} = \begin{cases} 1, & \text{如果聚合树 } t_k \text{ 包含边 } (u, v) \\ 0, & \text{其他} \end{cases} \quad (5)$$

对每一条边  $(u, v)$ , 可得以下波长约束:

$$\sum_{T_k \in T} \delta_{uvk} \leq w_{uv} \quad (6)$$

设  $T = \{t_1, t_2, \dots, t_k, \dots\}$  为聚合树,  $t_k$  表示第  $k$  棵聚合树中根和叶子的集合,  $T^n = \{t_1^n, t_1^n, \dots, t_k^n, \dots\}$  为组播组按照一定算法计算的生成树,  $t_k^n$  中每条边组播数据流请求的带宽之和为  $ba(t_k^n)$ , 与该组播组数据流对应的聚合树为  $t_k$ , 每条边传输的带宽之和为  $ba(t_k)$ ,  $ba(t_k)$  和  $ba(t_k^n)$  的差定义为  $d(t_k^n)$ , 对所有的  $t$  和  $t^n$ , 浪费的总带宽为  $d(t^n)$ . 如果  $d(t_k^n) = 0$ , 则表示完全匹配; 否则, 表示遗漏匹配,  $t$  和  $t^n$  满足以下约束:

- 头约束: 组播树  $t_k^n$  的头节点不能为空,  $t_k^n$  中所有元素必须与头节点相连.
- 带宽约束: 一棵聚合树上组播数据流带宽之和不能超过传输能力.
- 尾约束: 组播树  $t_k^n$  的尾节点不能为空, 即聚合组播组必须包含接收者.

组播聚合树的建立问题可描述如下:

假定图  $G(V, E, W)$ , 对给定的带宽  $B$  和组播组生成树  $t^n$ , 查找最优的聚合树  $t$ , 满足以上约束. 并使得:

- 聚合树的总数最小;
- 带宽浪费  $d(t^n)$  最小;
- 建立聚合树的总代价  $\sum \sum \delta_{uvk} \cdot w_{uv}$  最小.

注意, 组播聚合树中目标 3 包含两个部分:

- 网络应该建立尽可能多的组播数据的聚合树;
- 传输相同组播数据流, 选择代价小的聚合树.

建立组播聚合树的基本思想是, 找到覆盖给定组播组子集的聚合树. 对给定组播组子集, 覆盖所有组的树数量最小化问题可转化为经典的最小集合覆盖问题. 参照文献[7]的证明, 该问题是一个完全 NP 问题.

## 2 双邻域查找算法

在网络规模较小时, 虽然 ILP 算法可以找到组播聚合问题的最优解, 但对规模较大的网络则很难解决. 因为对有几个组播组的网络来说, 组播聚合问题为完全 NP 难问题. 因此提出一种双邻域查找算法, 使其适用于规模较大的问题. 该算法是一种启发式算法, 减小其复杂度, 但同时性能有所下降. 在本文的研究中, 双邻域查找算法不仅对静态有效, 而且对动态的加入和离开同样进行了处理.

双邻域查找算法基于拟物拟人等启发式算法, 主要包括基于局部查找的贪心算法生成聚合树和跳坑策略两部分. 贪心算法是一种最直接的设计技术, 它可被应用于多种问题, 基于贪心算法生成聚合树, 其基本法则是改变组播组生成聚合树的顺序, 以便减少聚合树的数量, 直到不能再减少聚合树数量为止. 由于以上组播聚合树生成解是基于局部查找算法, 只是在一个很小的局部区域中选取好的解, 如果能够找出更好的解就继续查找, 否则就停止. 它们几乎没有“向远处看”的策略, 即查找的全局观, 因此往往陷入局部最优的陷阱. 为了避免这种情况, 本文算法结合拟人拟物思想, 提出了跳坑策略.

在双邻域查找算法中,首先对网络中所有组播组按照组播路由协议生成组播树,然后采用双邻域查找算法对所有组播组的头节点和尾节点进行聚合,生成聚合树.如果聚合树不能映射到网络中相应波长进行传输,则需增大带宽浪费门限,将更多的组播组聚合到一棵聚合树上传输;如果聚合树数量小于可用波长的数量,则可适当减少带宽浪费门限.

完成组播聚合树的生成以后,将每一棵组播聚合树分配到相应的波长进行传输.本文中假定所有波长传输能力相同,且所有网络节点支持所有波长传输.

## 2.1 聚合方法

**聚合组播树.** 对网络中一个或多个组播组共享一棵组播树,进行多组组播数据的分发,这种多个组播组共享的组播树称为聚合组播树,简称聚合树,符号为  $t_i$ ,其中,  $i$  为聚合树下标.

**原始组播树.** 对网络中一个组播数据流包含的头终端节点、尾终端节点和通过某种组播路由由协议计算的中间传输节点,使用组播算法计算形成一棵组播树,称为原始组播树  $t_k^n$ ,其中,  $k$  为组播树下标,简称组播树.

将组播组聚合到聚合树的方法如下:

将组播组的头节点和尾节点合并到聚合树  $t_m$  时,其附加的链路不能超过  $l$  条,其中,  $l$  由原始组播树链路数量和带宽浪费门限来确定.原始组播树链路数量记为  $(|t_k^n| - 1)$ ,带宽浪费门限为  $b_{th}$ ,链路数  $l$  按下式计算:

$$l = (|t_k^n| - 1) \times b_{th} \quad (7)$$

原始组播组的头、尾节点和组播聚合树  $t_i$  进行聚合时,本文算法反复检查组播组中每一个节点  $v$ ,如果不在聚合树  $t_i$  中,对  $t_i$  进行扩充,包含  $v$  后的聚合树为  $t'_i$ ,如果扩充节点  $v$  后,聚合树  $t'_i$  对已经聚合的任意组播树附加链路不超过  $l$ ,则该聚合树可以扩展,包含节点  $v$ .像这样对原始组播组每一个节点进行处理.如果聚合树扩展后可以包含全部原始组播组节点,同时,新的聚合树  $t'_i$  与原始组播组满足扩展链路数小于  $l$ ,则将聚合树中所有链路消耗的带宽加上该组播组的需求带宽.如果存在链路带宽超过波长传输能力,则不能进行扩展;如果每一条链路传输的组播组带宽需求之和满足带宽约束,则将聚合树进行扩展,扩展后的聚合树为  $t'_i$ ,包含该原始组播组.

在本文的算法中,所有聚合方法均按照以上方法进行.

## 2.2 初始聚合树

对一个新的空聚合树集合,使用以下 3 个规则选择第 1 个组播组作为聚合树:

**规则 1(组播组对应的组播树链路数最少优先).** 组播树中所有链路数之和最小的组播树优先选择.

**规则 2(最少的尾接收优先).** 对于所选择的组播组,需要接收的尾终端最小.

**规则 3(组播组编号小优先).** 如果前面两个规则完全一样,则优先选择组播组编号小.组播组编号和实际应用中的组播地址通过哈希算法进行对应.

使用这 3 个规则,直观上保证选择的组播组与其他组播组聚合的可能性较大,使初始聚合树数量尽可能小.给出初始聚合的算法如下:

1. 在初始格局下,所有组播组的聚合树都没有分配.
2. 从旧格局向新格局演化时,按组播组优先选择规则在所有未聚合组播组中选取最先聚合的组播组.假设选取的组播组对应的组播树节点数为  $b$ ,从节点数范围为  $[b-l, b+l]$  的聚合树中逐一选取聚合树进行聚合.如果存在聚合树和选取的组播组满足聚合条件,则进行聚合;如果不存在满足聚合条件的聚合树,则将选取的组播组对应的组播树拷贝到聚合树集合中,成为一棵聚合树.

这样一步步地指定完所有原始组播组对应的聚合树,从而产生了初始聚合树.

## 2.3 邻域的构造

对于聚合树生成问题,算法按照先后顺序对每个原始组播组进行聚合,假定  $S$  是进行所有组播组  $g$  聚合的先后顺序序列,  $M_s$  是其中一棵聚合树中组播组聚合的先后顺序系列集合,  $M_s = \{S_k | S_k \text{ 是第 } k \text{ 棵聚合树中所有组播组进行聚合的先后顺序系列}\}$ .

**贪心方法.** 每次进行组播组聚合时,总是首先在已有聚合树中进行聚合,使总的聚合树数量保持最小值.在对聚合树中的组播组聚合顺序进行调整时,尽可能地使该组播组移出聚合树.在全部移出后,实现取消聚合树,从而减少聚合树的数量.这是一种贪心的策略.

**定理 1.** 对于同一网络中的组播组集合,给出的不同组播组聚合顺序系列  $S_1$  和  $S_2$ ,如果相对于每棵聚合树中的组播组被聚合时的先后顺序相同,则给出的不同系列  $S_1$  和  $S_2$  对应着相同的聚合树.

用数学归纳法可以很容易地证明以上定理.

**组播组调整策略.** 一棵按照贪心法给出的聚合树,除第 1 个组播组外的每一个组播组扩展的节点数进行排序,对扩展节点数最大的组播组进行调整,该组播组称为调整组播组.如果聚合树由单个组播组组成,则无须调整;如果有多个调整组播组,则随机选取 1 个组播组进行调整.调整的组播组为该聚合树差别最大的组播组.

**邻域结构 1.** 对于给定的组播组聚合序列  $S$ ,对其中一棵聚合树  $t_k$  中的组播组聚合序列  $M_s$ ,计算出调整组播组.假定组播组  $g_i$  需要进行调整,组播组  $g_i$  在聚合过程中的序号为  $i$ ,对该聚合树另一聚合过程中的任意序号  $j$ ,对应的组播组为  $g_j$ .在聚合树  $t_k$  中, $i$  在  $j$  之前,把  $i$  的位置移动到  $j$  的位置之后称为向后插入,把  $j$  的位置移动到  $i$  的位置之前称为向前插入.通过执行不同的向前或向后插入操作,形成新的组播组聚合序列  $S'$ ,则  $S'$  是  $S$  的邻解.

调整组播组进行向前插入或向后插入时,包含两种情况:假设聚合树  $t_k$  的第  $i$  个组播组  $g_i$  为调整的组播组,首先进行将调整组播组  $g_i$  向前插入到聚合树  $t_k$  第 1 个组播组之前,计算其邻解;然后将调整组播组  $g_i$  向后插入到聚合树  $t_k$  最后一个组播组之后,计算其邻解.如果以上两种调整不能减少聚合树数量,则随机选取位置进行向前或向后插入,计算其邻解.

邻域结构 1 的基本思想是,移动调整组播组,改变它们在聚合树中的位置,实现该组播组移出该聚合树.这样逐步减少聚合树的组播组,最终实现取消该聚合树,从而减少聚合树数量.

**邻域结构 2.** 对于组播组聚合顺序序列  $S_1$ ,按照聚合方法进行聚合的聚合树生成顺序为  $t_1, t_2, \dots, t_k$ .如果  $i < j$ ,则将  $t_i$  和  $t_j$  对应的组播组位置全部互换,生成新的组播组聚合顺序序列  $S'_1$ .由  $S'_1$  生成的聚合树  $t'_1, t'_2, \dots, t'_m$  称为  $t_1, t_2, \dots, t_k$  的邻解.

**定理 2.** 按照组播组调整策略给出的聚合树集合一定包含组播组聚合问题的最优解.

证明:任意给定组播组聚合顺序序列  $S_1$ ,对所有组播组的聚合顺序序列进行调整,调整完成后的顺序序列为  $S_2$ .因为调整的顺序按照贪心的方法对每一个组播组进行调整,所以  $S_2$  是一个按照贪心方法来聚合所有组播组得到的聚合序列.对于任意给定的聚合序列,按照贪心法对同一问题重新给出聚合顺序后,新的聚合顺序得到的聚合树数量一定小于或等于给出聚合序列的聚合树数量.因此,按照贪心法给出的聚合序列一定包含组播组聚合问题的最优解.  $\square$

## 2.4 邻域查找策略

**聚合树的对称聚合顺序.** 对任意给定的聚合树聚合顺序序列  $S_1$ ,将其第 1 个组播组修改为倒数第 1 个组播组,将其第 2 个组播组修改为倒数第 2 个组播组, ..., 将其倒数第 1 个组播组改为第 1 个组播组.由此形成的一个新的聚合顺序  $S'_1$  称为原聚合树的对称聚合顺序  $S'_1$ .

先计算原始聚合顺序  $S$ ,得到原始聚合顺序生成的聚合树集合  $t_1, t_2, \dots, t_m$ .再计算原始聚合顺序的对称聚合顺序  $S'$ ,得到对称聚合顺序生成的聚合树集合  $t'_1, t'_2, \dots, t'_n$ .取  $m$  和  $n$  中较小的那个聚合顺序作为较优的聚合顺序.

在邻域查找的过程中,只要发现邻域中某个聚合顺序  $S'$  对应的聚合树数量小于聚合顺序  $S$  的聚合树数量,就停止计算邻域中的其他聚合顺序,选定刚才所计算的聚合顺序作为下一次邻域查找的出发点.如果查找完邻域中所有的邻解都没有使聚合树数量小于原始出发点所对应的聚合树数量,则进行跳坑.

## 2.5 跳坑原则

跳坑是指查找最小聚合树数量陷入局部极小值后,按照某种策略随机选择一个聚合顺序作为新的邻域查找的出发点,这种策略称为跳坑策略.

在算法使用第 2 种邻域结构查找后,如果在邻域中找不到某个聚合顺序的聚合树数量小于对应聚合顺序

生成聚合树数量,则随机选取两棵聚合树,将其聚合顺序修改为其对称聚合顺序.由此形成新的聚合顺序序列.新的聚合顺序序列生成的聚合树必须有所改变.将此新的聚合顺序序列作为新的邻域查找的出发点.

除了以上跳坑方法以外,还采用随机同聚合树聚合顺序移动的跳坑方法.具体方法是:随机选择一棵聚合树中一半数目的组播组,先后随机移动它们的聚合顺序位置.移动完成后,再随机选取另外一棵聚合树做同样的动作,就完成了—次试跳.如果试跳后得到的聚合树数量不超过跳坑前聚合树数量的 110%,试跳被接受,得到一个新的聚合顺序作为下一次邻域查找的出发点.如果在算法设定的试跳次数内找不到这样的聚合顺序,则在所有的试跳中选择一个最好的聚合顺序进行跳坑,作为下一次邻域查找的出发点.

跳坑方法从历史最好解开始起跳,目的是希望能够继承一些聚合树不错的调度,能够更快地找到最优解.在跳坑之前,首先随机地从两种跳坑策略中选取 1 种,然后进行跳坑.每种跳坑策略的概率均为 0.5.

## 2.6 算法实现与描述

本节先说明邻域查找子程序和聚合树查找子程序两种算法.

邻域查找子程序描述如下:

在调用邻域查找子程序之前,当前聚合顺序的聚合树数量为  $k$ ,希望用邻域结构 1 的查找子程序找到聚合树数量  $m$  比  $k$  小的聚合顺序序列.设置一个数组  $Q_u$ ,数组  $Q_u$  中记录了最近访问过的聚合顺序序列生成的聚合树数量.如果某个邻解的聚合树数量在数组  $Q_u$  中,则称此邻解为被禁止的邻解.迭代次数用计数器  $H$  表示.邻域查找子程序描述如下:

- 1) 令  $H=0, Q_u$  中数值全为 0,当前顺序序列聚合树数量为  $k$ ,转步骤 2).
- 2) 考察当前聚合顺序序列的邻域结构 1 中的调度顺序,如果某个邻解的聚合树数量  $m$  小于当前顺序序列聚合树数量  $k$ ,那么接受这个邻解,停止邻域查找子程序.
- 3) 如果每个邻解的聚合树数量都大于等于当前顺序序列的聚合树数量,那么考察不被禁止的邻解,找到聚合树数量最小、第二小和第三小的邻解,在这 3 个邻解中均匀随机地选取一个,接受此邻解,并将此邻解的聚合树数量记录在数组  $Q_u$  中.
- 4)  $H=H+1$ .
- 5) 如果  $H>100$ ,则回到调用邻域查找子程序之前的那个聚合序列,停止邻域查找子程序;否则,转步骤 2).

聚合树查找子程序描述如下:

当前聚合顺序的聚合树数量为  $n$ ,聚合树查找子程序的描述如下:

- 1) 令  $k=1$ .
- 2) 如果  $k=n+1$ ,则停止聚合树查找子程序;否则,反复调用邻域查找子程序,直到不能优化聚合顺序为止.
- 3) 如果找到了比在调用聚合树查找子程序之前的那个聚合顺序生成的聚合树数量更少的聚合顺序,则停止聚合树查找子程序;否则,回到调用聚合树查找子程序之前的那个聚合顺序.
- 4)  $k=k+1$ ,转步骤 2).

双邻域查找算法描述如下:

- 1) 按照初始聚合树生成方法产生初始聚合顺序.
- 2) 判断是否满足停止条件.若满足停止条件,则给出历史最好的聚合顺序作为最终聚合顺序,转步骤 6).
- 3) 计算聚合树查找子程序中的解.调用聚合树查找子程序,如果找到了比当前聚合顺序生成的聚合树数量少的聚合顺序,则将此邻域结构聚合顺序作为当前聚合顺序,转步骤 2);如果不存在这样的聚合树,则转步骤 3).
- 4) 随机选取两棵聚合树,将其对应的组播组顺序互换,转步骤 3),计算邻域结构 2 中的解.若一旦查找到邻域 2 中的某个调度顺序的聚合树数量小于当前调度顺序的聚合树数量,则将查找到的聚合顺序作为新的出发点,转步骤 2).
- 5) 跳坑.如果已跳坑次数不能被 20 整除,那么以当前聚合顺序为起跳点;否则,以历史计算的聚合树数量最少的聚合顺序为起跳点,在两种跳坑策略中随机选取一种,然后跳坑,转步骤 2).

- 6) 将聚合树映射到传输波长上,如果存在链路的波长数量小于需要传输的聚合树数量,则将带宽浪费门限乘 2,作为新的带宽浪费门限,转步骤 1)重新计算.在带宽浪费门限增加的过程中,增加后的带宽浪费门限必须小于上限,如果带宽浪费门限达到上限,但聚合树数量仍大于波长数量,则随机选取波长数量的聚合树进行映射并传输,对未选中的聚合树进行丢弃.如果聚合树数量小于可用波长数量的一半,则将带宽浪费门限除以 2,转步骤 1)重新计算.在带宽浪费门限减小的过程中,减小后的带宽浪费门限必须超过下限;否则,接受本次聚合计算.

算法第 2 步中的停止条件为找到最优聚合树,或者跳坑次数达到 500 次.

## 2.7 动态加入和离开

假定静态组集合为  $G$ ,首先按照双邻域查找算法进行聚合,并将聚合后的聚合树分别分配到一个波长进行传输.当有组播组的加入和离开时,如果需要增加新的聚合树,则需重新调用双邻域查找算法进行计算.下面说明组播组的加入和离开过程:

- 加入过程:当一个新的组播组  $g$  加入到网络中时,首先判断是否存在合适的聚合树能够完全覆盖该组,对每一棵存在聚合树  $t_i$ ,如果需要扩展小于  $l$  条链路可以覆盖该组,同时满足约束条件,则可进行扩展.注意,对一棵存在聚合树进行扩展时,必须检查扩展后聚合树是否满足组原始组播组的  $b_{in}$  需求,以便保证带宽浪费是有效的.如果组播组  $g$  可以聚合到多棵聚合树中,选择最小带宽浪费树为最终聚合树;如果没有合适的聚合树可以覆盖该组播组,则计算该组播组对应的组播树,重新调用双邻域查找算法,进行聚合树的分配.
- 离开过程:当一个组播组离开时,找到该组对应的聚合树,在该聚合树中所有链路减去该组播组链路带宽需求.如果剩余组播组为空,则删除该聚合树;如果剩余组播组不为空,则对该聚合树重新进行聚合计算,形成新的聚合树,替换原来的聚合树.

在以上加入和离开过程中,重新进行双邻域查找时,可使用前期的运算结果,这样可以大幅度减少计算量.

## 2.8 复杂度分析

在生成聚合树算法中,对一个组播组  $g_i$ ,需要对组中每一个扩展节点进行扫描,判断是否满足带宽浪费门限,其复杂度为  $O(n^3)$ ,其中,  $n$  为节点数.每一个组播组的聚合中,迭代次数为常量  $k$ ,对每一个组播组最多需要与所有聚合树进行聚合计算,因此,总的复杂度为  $O(k \times N_g \times N_a \times n^3)$ ,其中,  $N_g$  为组播组数量,  $N_a$  为聚合树数量.

其运算速度远快于 ILP 算法的  $O(2^n)$ ,即便是对较小的带宽浪费门限,文献[7]中给出的组播聚合算法的时间复杂度为  $O(n^5)$ ,双邻域查找算法时间复杂度有较大的改进,且对带宽浪费门限没有限制,避免陷入局部最优,更容易得到最优解.

## 3 实验与分析

采用 NS2 仿真器在 PC 机进行算法的仿真实验.网络拓扑图如图 2 所示,该网络核心交换部分包含 20 个汇聚/核心路由器和 36 条链路,接入部分 20 个终端路由器,通过 20 条链路接入,每个节点的路由器都可以作为组播组转发的成员.

随机选取 6 个终端路由器为组播源头终端,其余终端路由器为尾终端,汇聚或骨干路由器中进行聚合树的聚合.4 000 个组播组从头终端向尾终端发送,在这些组播组中,60%需要 25Mb/s,30%需要 15Mb/s,其他 10%需要 5Mb/s.所有组播组均匀分布在 60%的头终端上,组播组请求按照泊松分布,连接保持时间按指数分布(平均值规格化统一单位).所有请求均匀分布在所有尾终端和组播组上.每一个尾终端接收组播组的平均数为 800,通过调节组播组的泊松分布参数,使得实验中并发的组播组数量始终维持在 4 000 个左右.每一条链路所有波长的带宽能力相同,如无特别说明,每一条链路支持的波长数量为  $\gamma=64$ .重复进行 100 次实验.初始组播树的生成通过最小代价生成树生成.双邻域查找带宽浪费门限下限为 5%,初始值取为 20%.

实验分为两个部分:首先,分别在每一个路由器上运行双邻域查找算法,对不同波长数量和波长传输能力条



件下,双邻域查找算法的各项性能;其次是针对不同的场景,分别在每一个路由器上运行双邻域查找算法,与文献[7]中的较经典伪动态组播聚合(pseudo-dynamic)算法和文献[6]中的启发式(heuristic)算法进行比较,测试不同算法的性能(不同场景分为轻载的空闲场景和重载的忙场景).

测试性能包含以下几种:聚合树的数量、平均带宽浪费比率和组播组阻塞率.

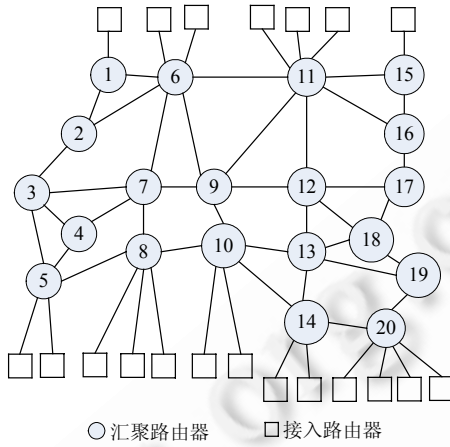
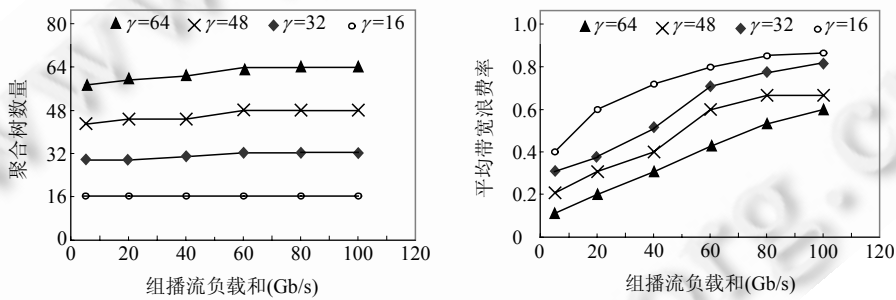


Fig.2 Topology graph of the simulation experiments

图2 模拟实验所用网络拓扑图

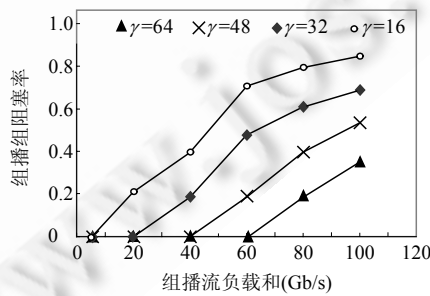
### 3.1 双邻域查找算法性能

实验中,每台路由器运行双邻域查找算法,图3说明组播流负载和与波长的平均数量 $\gamma$ 对聚合结果的影响.



(a) 聚合树数量与组播流负载的关系

(b) 平均带宽浪费率与组播流负载的关系



(c) 组播组阻塞率与组播流负载的关系

Fig.3 Relation between performance and multicast flow load

图3 性能与组播流负载的关系

从图 3(a)可以看出,随着 $\gamma$ 的增加,聚合树数量越来越多.这是因为双邻域查找算法中聚合树数量根据可用波长数量生成,随着 $\gamma$ 的增加,聚合树数量增加,即建立更多的聚合树,以便减少尾终端浪费的比率.如果保持可用波长数量不变,随着组播流负载和的增加,聚合树的数目基本保持不变.当波长数量较多、组播流负载和较低时,聚合树数量略少于波长数主要是由于组播组数目相对较少的原因.

图 3(b)说明了随着组播流负载和的增加,平均带宽浪费比率增加.其原因在于,组播流负载和增加后,形成部分组播组进行聚合,尾终端收到无效数据增多,导致尾终端浪费增加.随着波长数量 $\gamma$ 的增加,平均带宽浪费比例减少,其原因在于,波长数量越多,最终形成的聚合树越多,每一棵聚合树聚合的组播组减少,无效终端节点少,因此尾终端浪费减少.

网络中带宽能力不足会导致部分组播组无法聚合,出现阻塞的组播组,图 3(c)说明了组播组阻塞的比率.从图中可以看出:在波长带宽能力满足组播组需求之前,基本上没有组播组的阻塞;随着组播组数量的增加,由于波长带宽能力的限制,开始出现阻塞;在相同组播组的情况下,随着波长数量的增多,阻塞越来越小.这说明双邻域查找对聚合树计算的有效性.

### 3.2 性能比较

为了比较邻域查找算法的性能,本节对邻域查找算法(double neighborhood search algorithm,简称 DNSA)与文献[7]提供的较经典伪动态组播聚合算法(pseudo-dynamic)和文献[6]提供的启发式算法(heuristic)进行比较,分别在每一个路由器上运行双邻域查找算法、伪动态组播聚合算法和启发式算法,仿真中包含 100 000 个节目的请求和离开操作,测试不同算法在不同场景下的性能.

#### 3.2.1 空闲场景

本节测试负载较低时的性能,网络中链路波长数量为 30,链路的传输能力为 1Gb/s.测试结果如图 4 所示.从图 4(a)中可以看出:当组播流负载之和小于 600Mb/s 时,平均带宽浪费比率较小;当组播流负载之和为 600Mb/s 时,平均带宽浪费比率开始出现增加,因为尾终端请求不同的组播组,需要聚合组播组到同一棵树上,导致平均带宽浪费比率增大.当组播流负载之和小于 600Mb/s 时,双邻域查找算法的平均带宽浪费比率为 0,而伪动态和启发式组播聚合算法存在带宽浪费.这是因为双邻域查找算法生成较多聚合树,尽可能地利用所有波长资源,较少进行聚合的原因,无多余的尾节点;而当组播流负载之和超过 600Mb/s 之后,双邻域查找算法的尾终端浪费比率只有其他两种组播聚合算法的 20%左右.这同样是由于双邻域查找算法保持尽可能低的门限,进行优化的聚合.

图 4(b)为轻载时,双邻域查找算法与其他两种算法的组播组阻塞比率.从图中可以看出:在轻载时,双邻域查找算法的组播组阻塞比率始终为 0;而其他两种算法即使在轻载时也存在组播组阻塞的情况.

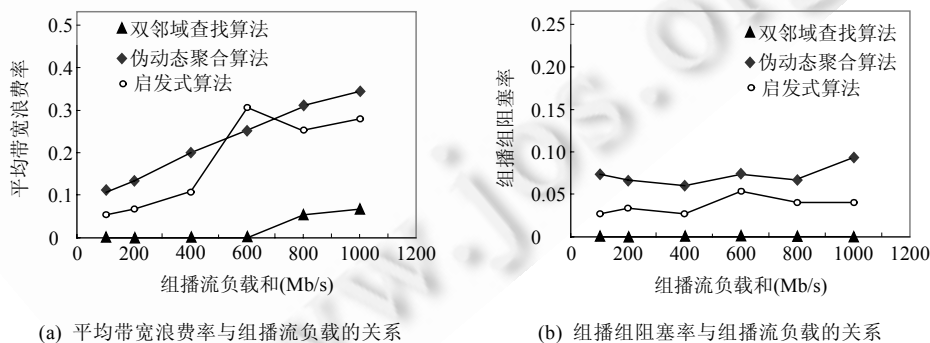


Fig.4 Performance comparison in low traffic load

图 4 轻载时性能的比较

#### 3.2.2 忙场景

在这个场景中,对负载较高的情况进行了测试.测试组播流负载之和为 100Gb/s~300Gb/s,网络中链路波长

数量为 100,网络中所有链路的传输能力为 1Gb/s.图 5 为实验结果.从图中可以看出,双邻域查找算法比其他两种组播聚合算法的平均带宽浪费比率至少要低 25%,同时,组播阻塞率小于 5%.这是因为双邻域查找算法生成的聚合树更接近最优解.从本场景可以看出:即使负载较重,使用双邻域查找算法也会显著减少聚合树的数量;而且双邻域查找算法只需 100 棵树即可满足需求,远远小于 IP 组播流所需要的树数量.

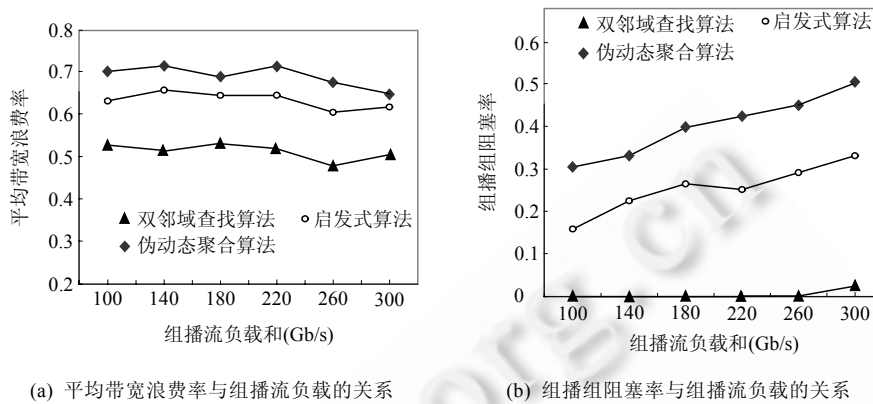


Fig.5 Performance comparison in high traffic load

图 5 重载时性能比较

从以上实验可以看出,使用聚合方法可以有效地减少网络中树的总数量,无论对网络轻载和网络重载,双邻域查找算法都可以获得较好的性能.

#### 4 结论

在光纤传输网络中,波长和带宽分配问题一直是困扰组播业务开展的致命问题.本文基于拟人的思想,提出了一种双邻域查找算法进行聚合.该算法采用不同的邻域结构,对组播组聚合过程中的顺序进行调整,使得局部查找具有更高的效率.而且在局部调整陷入极小值时,通过一系列的跳坑策略,使得寻找聚合树最优解走向更好的区域.另外,本文将聚合树和波长直接进行映射,避免了聚合树的映射过程.同时,根据光纤传输网络中的波长和带宽资源限制,对生成的聚合树数量和带宽浪费进行调整,实现最大限度的利用网络资源.仿真实验结果表明,双邻域查找算法可以有效地将网络中组播组进行聚合,减少网络中树的数量,并能节省带宽.另外,无论网络中负载为轻载或者重载,双邻域查找算法都具备较好的性能.

#### References:

- [1] Abhijit S, Frank S, Joe M, Julio N, Michael EA, Paolo D. Extending the service for successful and sustainable IPTV services. *IEEE Communications Magazine*, 2008,46(8):96–103. [doi: 10.1109/MCOM.2008.4597111]
- [2] Zhu Y, Jin Y, Sun W, Guo W, Hu W, Zhong WD, Wu MY. Multicast flow aggregation in IP over optical network. *IEEE Journal on Selected Areas in Communications*, 2007,25(5):1011–1021. [doi: 10.1109/JSAC.2007.070613]
- [3] Luis HC, Serge F, Otto CD. Incremental service deployment using the hop-by-hop multicast routing protocol. *IEEE/ACM Trans. on Networking*, 2006,14(3):543–556. [doi: 10.1109/TNET.2006.876157]
- [4] Ran LH, Rami M. Multicast routing and wavelength assignment in multi-hop optical networks. *IEEE/ACM Trans. on Networking*, 2002,10(5):621–629. [doi: 10.1109/TNET.2002.803907]
- [5] Cui JH, Kim J, Maggiorini D, Boussetta K, Gerla M. Aggregated multicast—A comparative study. *Cluster Computing*, 2005,8(1): 15–26. [doi: 10.1007/s10586-004-4433-8]
- [6] Rachana M, Pavel G, Arunabha S. A unified approach for multiple multicast tree construction and max-min fair rate allocation. In: *Proc. of the Int'l Conf. on High Performance Switching and Routing*. 2009. 1–8. [doi: 10.1109/HPSR.2009.5307420]

- [7] Cui JH, Lao L, Gerla M. A study of group-tree matching in large scale group communications. In: Proc. of the Int'l Symp. on Performance Evaluation of Computer and Telecommunication Systems. 2005. 1-8.
- [8] Cui JH, Faloutsos M, Gerla M. An architecture for scalable, efficient, and fast fault-tolerant multicast provisioning. IEEE Network, 2004,18(2):26-34. [doi: 10.1109/MNET.2004.1276608]
- [9] Huang XD, Farahmand F, Jue JP. Multicast traffic grooming in wavelength-routed WDM mesh networks using dynamically changing light-trees. Journal of Lightwave Technology, 2005,23(10):3178-3187. [doi: 10.1109/JLT.2005.856244]
- [10] Massimo T, Guido M, Achille P. WDM network design by ILP models based on flow aggregation. IEEE/ACM Trans. on Networking, 2007,15(3):709-720. [doi: 10.1109/TNET.2007.893158]
- [11] Neal C, Vinod MV. Static routing and wavelength assignment for multicast advance reservation in all-optical wavelength-routed WDM networks. IEEE/ACM Trans. on Networking, 2012,20(1):1-14. [doi: 10.1109/TNET.2011.2175007]
- [12] Seisho Y, Koji S, Ichiro I, Shigeo U. Scalable multicast MPLS protocol for next generation broadband service convergence network. In: Proc. of the IEEE Int'l Conf. on Communications. 2004. 1019-1023. [doi: 10.1109/ICC.2004.1312655]
- [13] Narendra KS, Laxman HS, Biswanath M. Optimal multicasting of multiple light-trees of different bandwidth granularities in a WDM mesh network with sparse splitting capabilities. IEEE/ACM Trans. on Networking, 2006,14(5):1104-1117. [doi: 10.1109/TNET.2006.882840]
- [14] Hsieh CY, Liao W. All-Optical multicast routing in sparse splitting WDM networks. IEEE Journal on Selected Areas in Communications, 2007,25(6):51-62. [doi: 10.1109/JSAC-OCN.2007.021305]
- [15] Yi Z, Jason PJ. Multi-Class flow aggregation for IPTV content delivery in IP over optical core networks. Journal of Lightwave Technology, 2009,27(12):1891-1903. [doi: 10.1109/JLT.2009.2022284]



汪学舜(1970-),男,湖北浠水人,博士,高级工程师,主要研究领域为流媒体数据传输,QoS保证.  
E-mail: wang\_xueshun@163.com



戴锦友(1968-),男,博士,高级工程师,主要研究领域为多业务以太网.  
E-mail: djiy96@fhn.com.cn



余少华(1962-),男,博士,教授,博士生导师,主要研究领域为IP互联网络,城域网.  
E-mail: shyu@fhn.com.cn