基于代理的覆盖网组播生成树算法研究*)

林龙新 周 杰 张 凌 叶 昭
(华南理工大学广东省计算机网络重点实验室 广州 510641)

摘 要 利用覆盖网组播技术构建组播服务平台是一种可行的提供组播服务的方案。基于代理的覆盖网组播兼具覆盖网组播的灵活性和IP组播的高效性的特点。结合节点的带宽、处理延迟和节点间的通信延迟给出一个完善的基于代理的覆盖网组播模型,根据此模型设计了求节点度受限的具有最小平均延迟的组播转发树生成算法。探讨了主机节点在进行数据分组复制转发时的转发顺序对平均延迟的影响,给出并证明了主机节点对数据分组复制转发的最优策略。通过仿真实验验证了所给算法和最优复制转发策略的有效性。

关键词 覆盖网组播,延迟,生成树

Research on Proxy-based Overlay Multicast Spanning Tree Algorithms

LIN Long-xin ZHOU Jie ZHANG Ling YE Zhao
(Guangdong Key Laboratory of Computer Network, South China Univ. of Tech., Guangzhou 510641, China)

Abstract The overlay multicast technology is being recognized as a feasible alternative to construct the general multicast service infrastructure. Proxy-based overlay multicast technology combines the feature of easy deployment of overlay multicast with high performance of IP multicast. By considering the node's bandwidth, process delay and communication delay between nodes, proposed a more appropriate overlay multicast model. Based on the model, a multicast tree construction algorithm with degree-bounded and minimum average delay was designed. Then the relationship between the average delay and the packet's forwarding orders was considered when a host forwards the copies of a packet to the downstream hosts, and an optimal forwarding strategy was proposed and proved. The simulation results show that the algorithm and optimal forwarding strategy are effective.

Keywords Overlay multicast, Delay, Spanning tree

1 引言

为解决IP组播部署困难的问题[1],研究者提出覆盖网(Overlay Network)组播技术[2]。当前的覆盖网组播技术主要分为两类:P2P结构的覆盖网组播(Peer to Peer architecture Overlay Multicast, P2POM)[3-5]和基于代理的覆盖网组播(Proxy-based architecture Overlay Multicast, POM)[6-8]。在P2POM中,组播服务功能完全由主机完成。在POM中,组播服务通过一些被称为组播服务节点(Multicast Service Node, MSN)的功能单元来完成,MSN是普通的主机或高性能专用服务器,用户(client)通过MSN来获得组播服务。

文献[9]对IP组播、P2POM和POM进行比较,结果表明,POM综合了IP组播和P2POM的优点,在保证组播性能的同时也容易部署,更适合构建通用的组播服务平台。本文只讨论POM的组播路由问题。文献[6,10]把POM中由MSN组成的覆盖网抽象成一个边赋权图,边的权表示MSN间的单播通信延迟。文献[10]把POM的组播路由问题转化为求度受限的最小直径生成树问题,目的是在满足带宽需求的同时,使MSN之间的最大通信延迟最小,同时给出了一个求度受限的最小直径生成树的集中式贪婪算法——CT(Compact Tree)算法。文献[6]把POM的组播路由问题转化为求度受限的最小平均延迟生成树问题,把POM中MSN所服务的client的数目考虑在内。文献[6,10]没有考虑MSN的处理延迟。由于MSN一般不具备线速转发能力,在实际应用中,其处理延迟不应被忽略。文献[11]给出的MDM(Minimum Delay Multicast)模型考虑了节点的处理延迟,但是并没有对生成树节点的度加以限制。

本文结合节点的带宽、处理延迟和节点间通信延迟,将由MSN组成的覆盖网抽象为一个节点和边都赋权的完全图G,将POM的组播路由问题转化为求G的节点度受限、具有最小平均延迟的生成树——MADDST(Minimum Average-Delay, Degree-bounded Spanning Tree)问题。设计了求节点度受限的最小平均延迟组播转发树生成算法。

在实际中,一个MSN一般只有一条接入链路,经过MSN复制转发的数据分组只能由物理链路顺序传输。在一棵组播转发树中,由于中间MSN复制转发数据分组的顺序不同,会导致数据分组到达其孩子节点的延迟不同,因而影响总体的平均延迟。合理安排中间MSN转发数据分组的顺序将使得总体平均延迟更优。给出并证明了节点对数据分组复制转发的最优策略。最后,通过仿真实验验证了所给算法和最优复制转发策略的有效性。

*)基金项目:国家"973"计划项目(2003CB314805),国家科技基础条件平台项目(2005DKA64001),2005年粤港关键领域重点突破项目"IPv6核心路由器研发与产品化"。林龙新 博士研究生,主要研究方向为应用层组播、对等网络;周 杰 副教授,博士;张 凌 教授,博士;叶 昭 博士研究生。

2 问题模型及算法描述

2.1 覆盖网通信模型

在实际的基于代理的覆盖网中，一个 client 可以通过直接给其提供服务的 MSN 来接收组播数据分组(图 1 中的用户 a,b,c)，也可以通过其他形式的覆盖网组播结构(例如 NICE[2]，图 1 中部分 I 的用户)或一个 IP 组播网络(图 1 的部分 II 的用户)间接获得组播服务。通过适当部署 MSN，可以形成一个世界范围的组播网络平台，使更多的 client 获得组播服务。图 1 为一个 POM 的组播转发树。

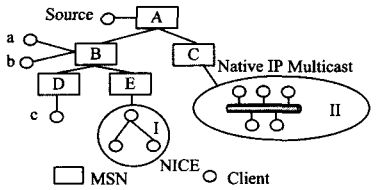

图 1 POM 的组播转发树

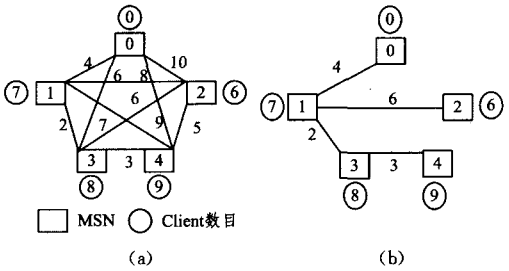

图 2 MADCT 算法生成的组播转发树

在 POM 中，所有 MSN 组成一个全连通虚拟网络，用无向完全图 $G=(V,E)$表示，其中 V 表示 MSN 的集合，E 表示 MSN 之间的单播路径。POM 中数据分组在 MSN 之间传输的延迟可分为两部分[11]：1) 通信延迟。数据分组在两个 MSN 之间的单播路径上的传输延迟，包括在单播路径上的累积传播延迟和累积排队延迟等。2) 处理延迟。MSN 处理一个数据分组(接收、复制和转发)的延迟。

对 $G=(V,E)$的任意节点 $u,v\in V,u\neq v$，边$\langle u,v\rangle\in E$ 的通信延迟用 $c(u,v)$表示。u 的处理延迟用 $p(u)$表示。假定在时刻 t，u 接收一个数据分组并发送给 v，u 在$[t,t+p(u)]$时间段内对数据分组进行处理。数据分组在时刻 $t+p(u)+c(u,v)$到达 v。如果 u 需向 $v_1,v_2,\cdots,v_k$ 转发同一数据分组的拷贝，且 u 只有一条接入链路，则 u 需顺序地将该分组的拷贝转发到其接入链路。因此先发送的数据分组拷贝在 u 中经历的处理延迟小于后发送的数据分组拷贝对应的处理延迟。文献[11]把 u 将数据分组的第 i 个拷贝转发到 v_i 的处理延迟定义为 $i\cdot p(u)$。事实上，数据分组在 u 中只经过一次接收、k 次复制和转发。因此，本文将组播服务节点 u 把数据分组的第 i 个拷贝转发到 v_i 的处理延迟定义为 $p_i(u)=w_0+i\cdot w(u)$，其中 w_0 是 u 接收分组的时间，$w(u)$是 u 复制和转发一个数据分组的时间。

2.2 度受限的最小平均延迟生成树问题

在一个由 MSNs 组成的覆盖网 $G=(V,E)$中，假定一个 MSN 节点 u 的接入带宽为 B_u，业务需求的带宽为 b，则 u 最多可以向$\lfloor B_u/b\rfloor$个其他的 MSN 节点发送数据分组。定义 $d_{\max}(u)=\lfloor B_u/b\rfloor$表示在组播转发树中 MSN 节点 u 的最大度。用 $a(u)$表示 u 所服务的用户数目。用 $D(r,u)$表示从组播转发树的根节点 r 到 u 的延迟。节点度受限的具有最小平均延迟的生成树问题(MADDST)为：

已知：一个无向完全图 $G=(V,E)$，对于任意节点 $v\in V$，最大度约束为 $d_{\max}(v)\in N$，节点处理延迟函数为 $p(v)\in Z^+$，v 所服务的端用户主机数目 $a(v)\in N$；对于任意边 $e\in E$，其通信延迟为 $c(e)\in Z^+$，节点 $r\in V$ 为根节点 。

求解：求 G 的一棵组播生成树 $T=(V',E')$，使得对任意 $v\in V'$，v 的度 $d_T(v)\leqslant d_{\max}(v)$，并且$\frac{1}{A}\sum_{u\in V}a(u)D_{r,u}$最小，$A$ 为所有端用户主机的总数目。

MADDST 问题是 NP-hard。因为假设 $p(v)=0,\forall v\in V$，即不考虑节点的处理延迟时，MADDST 问题等同于文献[6]定义的最小平均延迟生成树问题，而后者为 NP-hard。

2.3 算法描述

为求解 MADDST 问题，设计了一种类似于 Prim 算法的集中式贪婪算法，称为 MADCT 算法(Minimum Average Delay Compact Tree)。算法描述如下：

输入：$G=(V,E)$；$c(u,v),\forall\langle u,v\rangle\in E$；$d_{\max}(v),\forall v\in V$；$p(v),\forall v\in V$；$a(v),\forall v\in V$；$r\in V$。

输出：生成树 T 使得平均延迟最小。

```
1. T=(V'={r},E'=∅);//开始时 T 只包含一个根节点 r;
2. t(r)=0;// 初始化节点 r 到所服务的用户的平均延迟
3. d(r)=0;// 初始化节点 r 的度;
4. foreach u∈V-{r} do
// 初始化节点 r 外其他 MSN 节点的参数 t(u),m(u),d(u)
5.      t(u)=(c(r,u)+p(r))/a(u);
6.      m(u)=r;d(u)=0;
7. while V'-V≠∅ do
8.   if t(v)= min_{u∈V-V'} t(u) then
// 把平均延迟最小的节点 v 加入到 T 中，调整对应参数
9.      V'=V'∪{v};E'=E'∪{⟨m(v),v⟩};
10.     d(m(v))←d(m(v))+1;d(v)←d(v)+1;
11.     foreach v∈V-V' do
// 修改未加入到树 T 中的其他节点的相应参数 t(v)和 m(v)的值
12.     t(v)←∞;
13.     foreach u∈V' do
14.        if d(u)<d_max
   and (t(u)×a(u)+p(u)+c(u,v))/a(v)<t(v) then
15.           t(v)=(t(u)×a(u)+p(u)+c(u,v))/a(v);
16.           m(v)=u;
17. return T
```

其中，$T=(V',E')$表示算法生成的组播转发树。算法使用两个变量 $t(v)$和 $m(v)$。$t(v)$表示通过 T 传输数据分组时，分组从根节点 r 到达 MSN 节点 v 所服务用户的平均延迟；当 $v\in V-V'$时，$m(v)$表示 v 在 V'中对应的节点，满足通过添加边$\langle m(v),v\rangle$使得 $t(v)$最小。初始状态下 T 只包含根节点 r，初始化 V 中除 r 之外的所有节点对应的参数 $t(v)$和 $m(v)$。每次迭代都从 $V-V'$中选择一个 $t(v)$值最小的节点加到树 T 中，然后再重新调整剩余节点的参数 $t(v)$和 $m(v)$的值。

例如图 2(a) 表示由 5 个 MSN 节点组成的覆盖网，假设所有节点的处理时延都相同且值为 2，每个节点的最大度相同且值为 3。假定 0 为根节点，按照 MADCT 算法构造组播转发树，最后生成的组播转发树如图 2(b)所示。MADCT 算法主要由 while 循环构成，其中包含了两层 foreach 嵌套循环语句，故 MADCT 算法总的时间复杂度为 $O(n^3)$。

2.4 节点的最优转发策略

由于一个 MSN 一般只有一条接入链路，经过 MSN 复制转发的数据分组只能由物理链路顺序传输。因此，由于中间 MSN 复制转发数据分组的顺序不同，会导致数据分组到达其孩子节点的延迟不同，从而影响总体的平均延迟。例如，对于图 2(b)的组播转发树，节点 1 可以有两种方式转发数据分组：1) 先向节点 2 转发数据分组，后向节点 3 转发数据分组；2)先向节点 3 转发数据分组，后向节点 2 转发数据分组。按照 2.1 节的定义，$p_i(1)=w_0+i\cdot w(1)$，$i=1,2$。因为 w_0 为常量，对平延迟的计算没有影响，为简便起见，取 $w_0=0$。另外，设对任意 $u\in V$，$w(u)=2$。对于数据分组转发方式(1)，平均延迟为：

$$\overline{D}=\frac{6\times7+14\times6+12\times8+17\times9}{6+7+8+9}=12.5 \tag{1}$$

对于数据分组转发方式(2)，平均延迟为：

$$\overline{D}=\frac{6\times7+16\times6+10\times8+15\times9}{6+7+8+9}=11.77 \tag{2}$$

由此，组播转发树的中间节点按照不同转发顺序转发数据分组会影响平均延迟。下面给出优化平均延迟的转发策略。首先，对于由 MADCT 算法获得的组播转发树 T 的节点 v，递归地定义其服务的用户的总数值 $\Gamma(v)$ 为：

$$\Gamma(v)=a(v)+\sum_{u\in child(v)}\Gamma(u) \tag{3}$$

即 $\Gamma(v)$ 表示通过组播转发树 T 的节点 v 直接或者间接获得组播服务的用户的数目。例如在图 2(b)的组播转发树中，$\Gamma(3)=8+9=17$，$\Gamma(2)=6$，$\Gamma(1)=\Gamma(2)+\Gamma(3)+7=30$。

从组播转发树 T 的叶子节点开始递归地计算节点服务的用户总数值 $\Gamma(v)$：每个叶节点 v 计算自己的 $\Gamma(v)$ 值，并将该值报告给其父节点。然后按照式(3)计算其服务的用户总数值，并将该值报告给其父节点。如此递归地计算，直到根节点。

定理 给定组播转发树 T，除叶子节点外的每个节点按照其孩子节点服务的用户总数值从大到小的顺序向其孩子节点进行数据分组转发，则通过 T 传输数据分组的平均延迟最小。

证明：首先设 T 的深度 $h=2$，T 由根节点 r 和 k 个孩子节点 $v_1,v_2,\cdots,v_k$ 组成，组播系统的用户总数为 A。根节点 r 把数据分组的第 i 个拷贝转发到 v_i 的处理延迟为 $p_i(r)=w_0+i\cdot W$。从而，通过 T 传输数据分组的平均延迟为

$$\begin{aligned}\overline{D}&=\frac{(p_1(r)+c(r,v_1))\times a(v_1)+(p_2(r)+c(r,v_2))\times a(v_2)+\cdots+(p_k(r)+c(r,v_k))\times a(v_k)}{A}\\&=\frac{kw_0}{A}+\frac{W}{A}\sum_{i=1}^{k}ia(v_i)+\frac{1}{A}\sum_{i=1}^{k}c(r,v_i)a(v_i)\end{aligned} \tag{4}$$

式(4)中的第 1 项和第 3 项由组播转发树决定，与根节点的转发次序无关，只有第 2 项与转发次序有关。显然，当且仅当 r 按照 $a(v_i)$，$i=1,2,\cdots,k$ 从大到小的顺序，即 r 按照其孩子节点服务的用户总数值从大到小的顺序向其孩子节点进行数据分组转发时，$\overline{D}$ 取最小值。假设对深度为 $h(2<h\leqslant K-1)$的组播转发树结论成立。现设 T 是一棵深度为 K 的组播转发树，T 的根节点 r 有 k 个孩子节点 $v_1,v_2,\cdots,v_k$，则这 k 个孩子节点分别为 T 的深度小于等于 $K-1$ 的子树的根节点。设通过这 k 棵子树传输数据分组的最优平均延迟分别为 $\overline{D}_1,\overline{D}_2,\cdots,\overline{D}_i,\cdots,\overline{D}_k$，则通过树 T 传输数据分组的平均延迟为：

$$\begin{aligned}\overline{D}&=\frac{(p_1(r)+c(r,v_1)+\overline{D}_1)\times\Gamma(v_1)+(p_2(r)+c(r,v_2)+\overline{D}_2)\times\Gamma(v_2)+\cdots+(p_k(r)+c(r,v_k)+\overline{D}_k)\times\Gamma(v_k)}{A}\\&=\frac{kw_0+\sum_{i=1}^{k}c(r,v_i)\Gamma(v_i)+\sum_{i=1}^{k}\overline{D}_i\Gamma(v_i)+W\sum_{i=1}^{k}i\Gamma(v_i)}{A}\end{aligned}$$

此等式中只有 $W\sum_{i=1}^{k}i\Gamma(v_i)$ 与根节点 r 转发数据分组的次序有关。显然，当且仅当 r 按照 $\Gamma(v_i)$，$i=1,2,\cdots,k$ 从大到小的顺序，即 r 按照其孩子节点服务的用户总数值从大到小的顺序向其孩子节点进行数据分组转发时，$\overline{D}$ 取最小值，定理得证。

例如对图 2(b)的组播转发树，节点 1 按照先向节点 3 转发数据分组，后向节点 2 转发数据分组时，通过树 T 传输数据分组的平均延迟最小(见式(2))。

3 实验测试

为验证 MADCT 算法的有效性，用 C++语言实现了 CT 算法[10]、MADCT 算法以及具有最优转发策略的 MADCT 算法（简称为 O-MADCT 算法)，对通过构造的组播转发树传输组播业务的平均延迟进行对比测试。同文献[11]，分别对全连接覆盖网拓扑结构和部分连接覆盖网拓扑结构进行测试。通过 GT-ITM[12]工具包产生 IP 网络和部分连接覆盖网拓扑结构，然后转换为 2.1 节定义的覆盖网络进行测试。

实验 1 全连接覆盖网拓扑结构下的比较测试。产生一定节点规模的无向完全图 G，设每个节点的最大度约束 d_{max} (v)都相同，且值为 4；在每次测试中，每个节点的处理延迟 $p(v)$、服务的用户总数 $a(v)$以及 G 中边所代表的通信延迟 $c(e)$用取自一定范围的随机整数来表示。记 $Z=(p(v),a(v),c(e))$。用 $Z=([a,b],[c,d],[e,f])$表示各参数的取值范围。例如，在试验中分别选取 $Z_1=([1,10],[1,100],[1,10])$，$Z_2=([1,1],[1,100],[1,10])$，$Z_3=([1,10],[1,100],[1,1])$，其中 Z_1 中的第一个分量[1,10]表示节点的处理延迟 $p(v)$的取值为 1 到 10 之间的一个随机整数。Z_1，Z_2 和 Z_3 分别表示节点的处理延迟和通信延迟基本接近，处理延迟远小于通信延迟，处理延迟远大于通信延迟 3 种情况。对于 CT 和 MADCT 算法，在生成组播转发树后，节点按随机次序向其孩子节点进行数据分组转发。O-MADCT 算法按照最优转发策略进行数据分组转发。测试结果如图 3 所示，图中的每个数据点所表示的数值是 30 次测试结果的平均值。

由图 3 可知，通过由 MADCT 和 O-MADCT 算法构造的组播转发树传输组播业务的平均延迟明显低于由 CT 算法构造的组播转发树，其中通过 O-MADCT 算法得出的平均延迟最低。当节点处理延迟较大时(图中 Z_1 和 Z_3 条件下，这更符合 POM 的实际情况)，通过由 O-MADCT 算法构造的组播转发树传输组播业务的平均延迟明显低于通过由 MADCT 算

法构造的组播转发树。

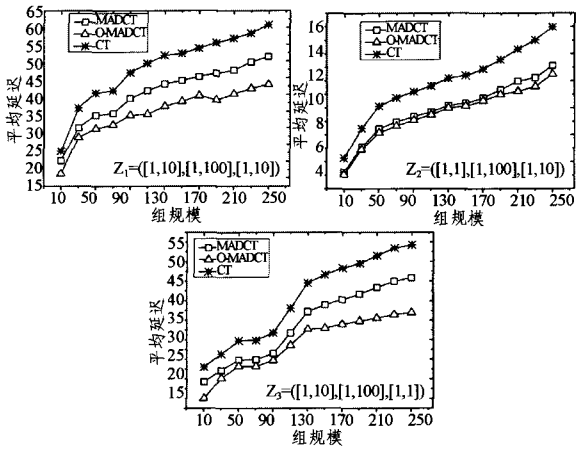

图 3　全连接覆盖网拓扑结构下的比较测试

实验 2　部分连接覆盖网拓扑结构下的比较测试。用 GT-ITM 产生 1000 个节点的随机图 G，设置参数“scale＝1000”，即把 G 放置到 1000×1000 的平面上，设置节点之间的连接概率 $\alpha=0.033$。从 G 中随机选取一定数目的节点作为覆盖网节点集合 V，并根据最短路径算法计算出 V 中每对节点的路径，把路径的长度（这里指跳数）作为对应边的通信延迟。设最大度约束 $d_{max}(v)$ 都相同，且值为 4；在每次测试中，设置每个节点的处理延迟 $p(v)$、服务的用户总数 $a(v)$（在我们的测试中，$a(v)$ 取[1，100]的任意整数值）。通过选取不同的 $p(v)$ 值进行测试。测试结果如图 4 所示。

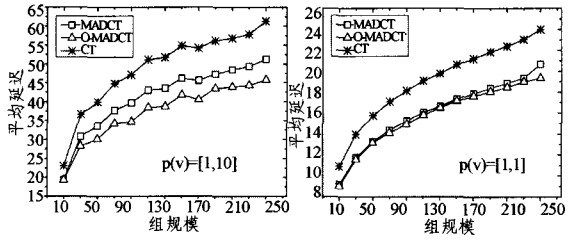

图 4　部分连接覆盖网拓扑结构下的比较测试

由图 4 可知，通过由 MADCT 和 O-MADCT 算法构造的组播转发树传输组播业务的平均延迟明显低于通过由 CT 算法构造的组播转发树传输组播业务的平均延迟，其中通过 O-MADCT 算法构造的组播转发树传输组播业务的平均延迟最低。

结束语　本文讨论了基于代理的覆盖网组播路由中具有最小平均延迟的组播转发树的生成算法。在已有覆盖网通信延迟模型的基础上，通过结合节点的带宽、处理延迟和节点间的通信延迟给出一个完善的基于代理的覆盖网组播模型，基于此模型设计了求节点度受限的具有最小平均延迟的组播转发树生成算法——MADCT 算法。在 MADCT 算法生成的组播转发树中，由于中间 MSN 复制转发数据分组的顺序不同，会导致数据分组到达其孩子节点的延迟不同，因而影响总体的平均延迟。给出并证明了节点对数据分组复制转发的最优策略。仿真实验表明，通过由 MADCT 算法构造的组播转发树传输组播业务的平均延迟明显降低。当 MSN 的处理延迟相对较大时，通过由 O-MADCT 算法构造的组播转发树传输组播业务的平均延迟明显低于通过由 MADCT 算法构造的组播转发树传输组播业务的平均延迟。

参 考 文 献

[1] Diot C, Levine B N, Lyles B, et al. Deployment issues for the IP multicast service and architecture. IEEE Network, 2000, 14(1): 78-88

[2] El-sayed A, Roca V, Mathy L. A survey of proposals for an alternative group communication service. IEEE Network, 2003, 17(1): 46-51

[3] Castro M, Druschel P, Kermarrec A M, et al. Scribe: a large-scale and decentralized application-level multicast infrastructure. IEEE Journal on Selected Areas in Communications, 2002, 20(8): 1489-1499

[4] Miguel C, Peter D, Anne-marie K, et al. SplitStream: high-bandwidth multicast in cooperative environments // Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles. New York, 2003

[5] Xinyan Z, Jiangchuan L, Bo L, et al. CoolStreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming // INFOCOM 2005. Miami, Florida, 2005

[6] Banerjee S, Kommareddy C, Kar K, et al. Construction of an efficient overlay multicast infrastructure for realtime applications // Proceedings of the IEEE INFOCOMM 2003. San Franciso, 2003

[7] Jannotti J, Gifford D, Johnson K, et al. Overcast: Reliable Multicasting with an Overlay Network // Proceedings of USENIX OSDI. San Diego, 2000

[8] Chawathe Y. Scattercast: an adaptable broadcast distribution framework. Multimedia Systems, 2003, 9: 104-118

[9] Lao L, Cui J H, Gerla M, et al. A Comparative Study of Multicast Protocols: Top, Bottom, or In the Middle // Proceedings of 8th IEEE Global Internet Symposium in Conjunction with INFOCOM'05. Miami, Florida, 2005

[10] Shi S Y, Turner J S. Multicast Routing and Bandwidth Dimensioning in Overlay Networks. IEEE Journal on Selected Areas in Communications, 2002, 20(8): 1444-1455

[11] Brosh E, Levin A, Shavitt Y. Approximation and Heuristic Algorithms for Minimum-delay Application-layer Multicast Trees. IEEE/ACM Transactions on Networking, 2007, 15(2): 473-484

[12] Zegura E W, Calvert K L, Bhattacharjee S, et al. How to model an internetwork // Proceedings of INFOCOM96. San Francisco, 1996