

# 针对组播的流量控制:带权的代价-收益法<sup>\*</sup>)

何炎祥 吕 慧 李旭晖 杜卓敏 王 策

(武汉大学计算机学院 软件工程国家重点实验室 武汉430072)

## Flow Control for Multicast: A Weighted Cost-Benefit Approach

HE Yan-Xiang LÜ Hui LI Xu-Hui DU Zhuo-Min Wang Ce

(School of Computer, Wuhan University, Wuhan 430072) (State's Key Lab. of Software Engineering, Wuhan University, Wuhan 430072)

**Abstract** This paper presents a weighted cost-benefit approach based on multicast. Grounded on overlay network model, its principal idea is to construct cost function and benefit function. When estimated benefit is more than cost, multicast request is accepted. This paper demonstrates its application in packet-switched networks and points out its merits and shortcomings.

**Keywords** Flow control, Overlay network model, Overlay network node, Cost-benefit approach, Weighted cost-benefit approach

## 1 引言

组播(multicast)是种一点对多点的传输方式,随着网络技术的发展,将应用于视频会议、视频点播、远程教育等各个方面。IP协议专门预留了组播地址。IPv4协议对组播预留了D类地址,一个IP组播组由一个D类地址标识,D类地址的高四位为“1110”,后跟28位组播组标识;IPv6协议也为组播预留了地址空间,地址高8位为“11111111”,后跟120位组播组标识。

流量控制是组播中的一个重要问题,通过限制发送方发出的数据流量,尽量使发送方发送速率不要超过接收方能处理的速率。目前,流量控制方案主要是滑动窗口协议。滑动窗口协议<sup>[1]</sup>的关键是:任何时刻发送过程都保持一组序列号,对应于允许发送但尚未确认的帧,这些帧称作在发送窗口内,当帧的确认到来,窗口上限、下限加1;当网络层的一个新分组到达,都会给此分组下一个最高的序列号。类似的,接收过程也维持一个接收窗口,对应于一组允许接收的帧。本文提出了一种适用于组播的带权的代价-收益法来进行流量控制。

## 2 代价-收益法流量控制基于的架构

本文描述的组播流量控制依赖于覆盖网络模型(Overlay network model)<sup>[1]</sup>,该覆盖网络模型是对网络拓扑结构的抽象,是由覆盖网络节点和虚拟链路构成的图。如图1所示为一组播系统,下面一层是网络的拓扑结构,上面一层是覆盖网络模型。在网络拓扑结构层,可以把组播系统看成是由多个小组组成,通常,主机作为一般节点和离它最近的特殊节点相连构成小组。特殊节点运行守护进程(daemon program),提供组播服务,兼作路由器,定期将本小组的状态以组播形式发送给其他覆盖网络节点,同时接收其他覆盖网络节点送来的状态消息,了解整个网络状态。特殊节点和覆盖网络节点一一对应。

网络拓扑结构可用图 $G=(V, E, \Phi, g)^{[2]}$ 表示,其中: $V=V_c \cup V_s$ ,  $V_c$ 是一般节点的集合,  $V_s$ 是运行守护进程的特殊节点集合;  $E=E_p \cup E_d$ ,  $E_p$ 是物理线路集合,  $E_d$ 是数据链路的集合;  $\Phi: E \rightarrow V \times V$ , 是从集合E到节点偶对上的函数;  $g$ 是小组的集合,定义如下:

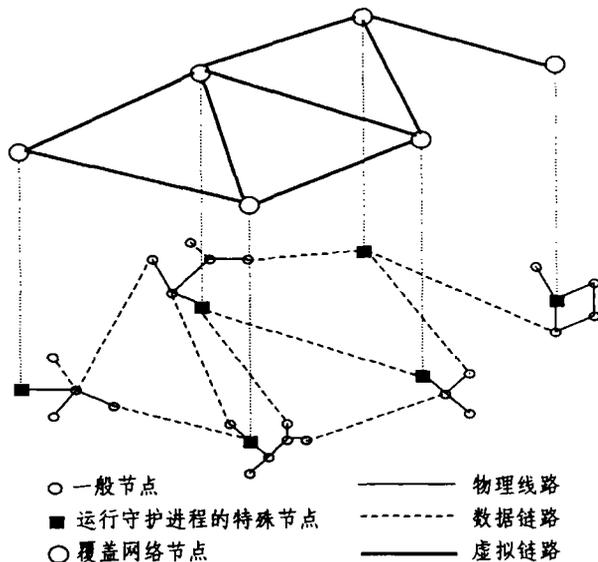
$$g = \{g_i | \exists (v_i \in g_i \wedge v_i \in V_s) \wedge (v_m \in g_i \wedge v_m \text{ 和 } v_n \text{ 间存在一条路径 } (e_1, e_2, \dots, e_k); (e_1, e_2, \dots, e_k \in E) \wedge k < K \rightarrow v_n \in g_i), i=1, 2, \dots, N\} (K \text{ 为给定的某个常数})$$


图1 网络拓扑结构及其相应的覆盖网络模型

覆盖网络模型基于网络拓扑结构,可用图 $G'=(V', E', \Phi')$ 表示,其中 $V'$ 是覆盖网络节点的集合,与运行守护进程的特殊节点集合一一对应,即存在一一映射 $f: V_s \rightarrow V'$ ;  $E'$ 是虚拟链路的集合,如果 $v_m \in g_i \wedge v_n \in g_i$ ,且 $v_m$ 和 $v_n$ 间存在一条链路 $e_k \in E$ ,则 $g_i$ 和 $g_i$ 间存在一条虚拟链路 $e_p \in E'$ ,且 $\Phi'(e_p)$

<sup>\*</sup>)本课题得到武汉市科技重点计划项目资助。何炎祥 博士,教授,博士生导师,主要研究方向为分布并行处理。吕 慧 博士,主要研究方向为分布并行处理,计算机网络。李旭晖 博士,主要研究方向为分布并行处理。杜卓敏 博士,主要研究方向为分布并行处理。王 策 硕士,主要研究方向为分布并行处理。

$= (v_s, v_r)(v_s, v_r \in V')$ ;  $\Phi: E' \rightarrow V' \times V'$ , 即是从集合  $E'$  到覆盖网络节点偶对上的函数。

目前,一些组通信系统能提供网络拓扑结构及其覆盖网络模型间的映射。

### 3 代价-收益法(Cost-Benefit Approach)

代价-收益法的基本思想是构造代价函数和收益函数。当覆盖网络节点获得一个组播请求时,计算相应的代价和收益,如果收益大于代价,则接受该请求,否则拒绝,从而来控制网络上的流量。对于网络来说,主要是资源的竞争,比如带宽、缓冲区等,可以根据具体情况选择我们最关心的资源作为衡量指标。

#### 3.1 代价函数和收益函数

代价函数用来衡量使用资源必须付出一定的代价。定义的代价函数<sup>[1]</sup>与资源利用率有关,而且随着资源利用率增加,代价也相应增加,当资源用尽时,任何组播请求都不能接受。代价函数  $C$  定义如下:

$$C: [0, 1] \rightarrow [\alpha, \beta], C(u_1) = \alpha \left(\frac{\beta}{\alpha}\right)^{u_1}$$

其中:  $u_1$  为资源 1 的利用率,  $0 \leq u_1 \leq 1$ ;  $\alpha$  为一个单位资源的最小收益;  $\beta$  为一个单位资源的最大收益。

收益函数用来衡量若接受组播请求,占用资源预计获得的一定收益,如发送者吞吐量、接收者吞吐量等。收益函数  $B$  定义如下:

$$B: [\alpha, \beta] \rightarrow [\alpha, \beta], B(\gamma_1) = \gamma_1$$

其中:  $\gamma_1$  为一个单位资源的收益,  $\alpha \leq \gamma_1 \leq \beta$ 。

代价函数与收益有关,随着资源利用率提高,代价呈指数增长。当资源未用即  $u_1 = 0$  时,代价  $C(0) = \alpha$ , 此时,如果接受一个组播请求,必定有预计收益  $B(\gamma_1) > \alpha$ , 收益大于代价,所以资源未用时可接受任何组播请求;当资源用尽时即  $u_1 = 1$  时,代价  $C(1) = \beta$ , 此时,必定有  $B(\gamma_1) \leq \beta$ , 收益不大于代价,所以资源用尽时,不接受任何组播请求。

构造的代价函数和收益函数仅计算树中某个节点上的代价和收益。通常,一个组播信息发送出去,要经过多个节点,每条组播信息在树中的每个分支上仅需一条,所以总代价  $C_t$  和总收益  $B_t$  分别为:

$$C_t = \sum_{i \in L_p} C(u_i) \quad B_t = \sum_{i \in L_p} B(\gamma_i)$$

其中:  $L_p$  为选择的组播路径。

当一个节点(包括覆盖网络节点)要进行组播时,首先向其所在小组的覆盖网络节点提出组播请求,覆盖网络节点根据一定策略<sup>[4]</sup>选择一棵树,该覆盖网络节点作为树根,根据代价-收益法进行判断,决定是提供该服务还是拒绝该服务,如果总收益大于总代价,则提供该服务,将组播信息发送到其它覆盖网络节点,由其它覆盖网络节点发送给一般节点。

#### 3.2 带权的代价-收益法

在实际应用中,不同类型的组播请求影响程度可能不一样,比如说网络状态信息比一般的数据信息要重要得多,或者一个组播请求多次被拒绝可能引起饥饿现象。代价-收益法同等对待所有的组播请求,通过在收益函数中加权,可以解决这个问题。

带权的收益函数  $B'$  定义如下:

$$B': [\alpha, \beta] \rightarrow [\alpha, \beta], B'(\gamma_i) = W\gamma_i$$

$W$  是权,  $W \in [1, \beta/\alpha + \theta]$ ,  $\theta$  是趋近于 0 又不等于 0 的正数,其值根据组播信息的类型确定,优先级高的信息其权比优先

级低的信息权高。当  $W = 1$  时,  $B'(\gamma_i) = B(\gamma_i)$ 。

当某个组播请求必须接受时,可以令权值  $W = \beta/\alpha + \theta$ , 此时带权的收益函数  $B'(\gamma) = \beta + \alpha\theta > \beta$ , 收益必定大于代价,组播请求被接受。当某个节点提出的组播请求被拒绝时,它下一次提出请求时,可适当增加其权,使其被接受的可能性不断增加。

### 4 带权的代价-收益法在分组交换网上的应用

在分组交换网上,我们比较关心的资源之一是路由器的缓冲区。路由器将到达的分组送到相应的输出线路,如果输出线路的缓冲区空,分组立即转发,否则在缓冲区中排队。当缓冲区满时,需要根据一定策略丢弃一些分组。

令路由器的缓冲区最大容量为  $M$ , 用  $x$  表示缓冲区中的信息量,则缓冲区的利用率  $u_b = x/M$ 。实际上,我们可以令一个单位资源的最小收益  $\alpha = 1$ , 实际收益的衡量以此为基础。设一个单位资源的最大收益是最小收益的  $Max_B$  倍,代价函数可简化为:

$$C: [0, M] \rightarrow [1, Max_B], C(x) = Max_B^{x/M}$$

对代价函数进行一个比例变换,使其值域在 0 到某个数  $S$  之间:

$$C: [0, M] \rightarrow [0, S], C(x) = S \cdot \frac{Max_B^{x/M} - 1}{Max_B - 1}$$

对带权的收益函数做相似的处理,可得:

$$B': [1, Max_B] \rightarrow [0, S], B'(\omega) = WS \cdot \frac{\omega - 1}{Max_B - 1}$$

其中:  $\omega$  为一个单位资源预计收益与最小收益的比值;  $W$  是权,  $W \in [1, \rho]$ ,  $\rho$  是比 1 大的正数。

### 5 分析与比较

带权的代价-收益法对资源定义价格和收益,根据价格和收益决定其操作,具有一定特点,与滑动窗口协议相比,也有其优点。

(1) 代价-收益法构造的代价函数和收益函数比较简单,容易计算。

(2) 代价-收益法利用网络当前的状态,计算代价和收益,动态调节网络流量。

(3) 带权的代价-收益法可以避免饥饿现象的发生,具有一定的公平性。

(4) 滑动窗口协议只有收到确认信息才能传送下一帧,而且一条组播信息有多条确认信息,增加了网络流量,较适合面向连接的传送;代价-收益法不需要等待反馈信息,适合无连接的传送。

(5) 在滑动窗口协议中,当一定时间内收不到确认消息,必须重传该帧,可能使网络拥塞更加严重;在代价-收益法中,代价函数呈指数增长,而收益函数是线性增长,当网络拥塞严重时,组播请求一般被拒绝。

(6) 滑动窗口协议同等对待所有帧,没有优先级;带权的代价-收益法通过对  $W$  赋不同的值,使组播请求具有不同的优先级别。

当然,代价-收益法也存在一定的缺点:

(1) 总收益和总代价主要衡量网络的整体情况,却不能描述局部情况,有可能预计的总收益较好,可实际上所选路径的中间节点发生了严重拥挤现象,实际收益并不等于预计值。

(2) 覆盖网络节点要定期发送网络状态信息,占用一定时间和资源。通过某些方法来尽量减少其占用时间和资源,如状

态发生较大改变才发送信息。

(3)不需要反馈信息,所以并不知道信息是否正确接收,较适合视频会议、视频点播等系统,帧丢失对用户来说影响不大。

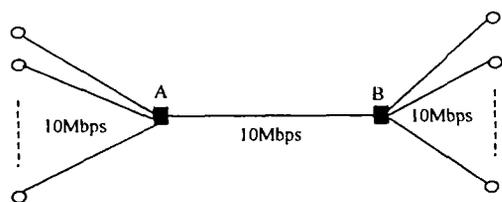


图2 网络拓扑结构

利用仿真器在如图2所示的网络拓扑结构上测试带权的

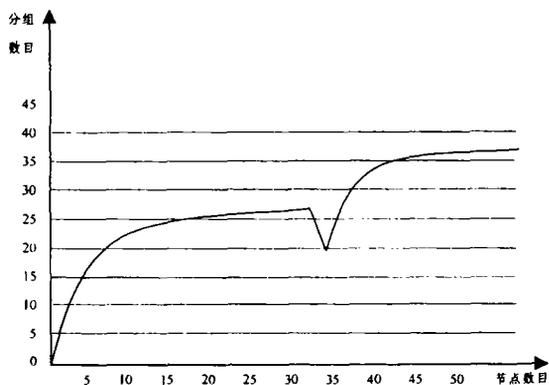


图3 发送节点数目与分组数目关系的坐标图

代价-收益法的性能,发送节点连到特殊节点 A,连续提出组播请求,接收节点连到特殊节点 B.限制  $M=30$ ,令  $M_{\max}=e$ ,一个组播请求被拒绝后重新请求其权加0.1,相应的代价函数和收益函数分别为:

$$C(x) = S \cdot \frac{e^{x/M} - 1}{e - 1} \quad B'(\omega) = WS \cdot \frac{\omega - 1}{e - 1}$$

发送节点数目与分组数目间关系如图3所示,当发送节点数目从0开始增长,分组数目迅速增长;随着发送节点数目接近30,分组数目保持平稳;发送节点数目超过30,分组数目突然减少然后缓慢增长,保持在40以下。

**结束语** 本文提出的带权代价-收益法,基于覆盖网络模型,由覆盖网络节点根据获得的网络状态信息以及请求的级别,计算相应的代价函数和收益函数,当总收益大于总代价时,执行组播。由于在进行判断前,覆盖网络节点要根据一定策略选择一棵树,策略的不同可能导致选择的树不同,从而得到的判断结果可能不同。该方法进行一些适当修改,同样能适用于单播和广播。该方法与负载平衡和树选择算法等技术综合起来应用,能进一步提高其性能。

## 参考文献

- 1 Amir Y, Awerbuch B, Danilov C, Stanton J. Flow Control for Many-to-Many Multicast: A Cost-Benefit Approach. <http://www.cnds.jhu.edu/publications/>
- 2 方世昌. 离散数学. 西安电子科技大学出版社, 1998
- 3 Andrew S. Tanenbaum. Computer Networks (Third Edition). Tsinghua University Public House, 2000
- 4 吕慧, 吴产乐, 周一勤, 韦峰. 多媒体远程教育网中的实时组播研究. 武汉大学学报, 1999, 45(5A): 531~534

(上接第117页)

·与 Jini 的比较 JXTA 和 Jini 都能用来实现分布式计算,都能用来连接分布式服务,都提供了网络中的计算机设备彼此发现的机制。但二者的不同也是显然的: JXTA 用以在一个对等网络中连接分布式服务,以 XML 格式定义了一组协议,而不是应用编程接口(API),因此 JXTA 就能够独立于编程语言、独立于操作系统,也独立于网络平台; Jini 技术则是在 Java 网络中连接分布式服务, Jini 假设 Java 字节码能够随处予以解释,由此 Jini 设计了一种基于移动 Java 对象以构建服务网络的方法,结果就使得 Jini 虽不依赖于协议但依赖于 Java 虚拟机,至少是网络上有一台设备能提供 Java 环境。

JXTA 与 Jini 也能共存。可以使用 JXTA 的协议以找到一个 Jini 查找服务(lookup service)、或找到一个 Jini 代理主机。一旦找到一个 Jini 代理主机, JXTA 网络中的用户就可以加载一个 Java 对象以作为连接 JXTA 网络和 Jini 网络的网关。

·与 HailStorm 的比较 相同的是,这两种技术都采用 XML 作为消息格式。但 HailStorm 的重点在于实现数字设备连接到中央服务器上的服务。而 JXTA 所在的网络中却不一定有中央服务器。而且 JXTA 是开放源代码的,有别于申请专利的 Hailstorm。此外,二者也存在同一性, HailStorm 中的护照技术(Passport)可嵌入 JXTA,实现一个认证功能。

**结束语** 针对当前 P2P 应用中面临的问题, JXTA 针对

性地提出了有效的解决机制,实现了互操作性、平台独立性、普遍性和安全性,极大地提高了带宽、中央处理器、存储系统、及信息资源的利用率。我们有理由相信,开放源代码的 JXTA 将使 P2P 开发人员免除复杂的开发工作,能够更便捷地创建出各类新颖、高效、创造性的分布式计算应用。

## 参考文献

- 1 Sun Microsystems. Peer-to-Peer Programming Using JXTA . Available at: <http://www.sun.com/developers/evangcentral/presentations/jxta.pdf>, 2001
- 2 Sun Microsystems. Project JXTA: A Technology Overview. Available at: <http://www.jxta.org/project/www/docs/TechOverview>. April 25, 2001
- 3 Sun Microsystems. PROJECT JXTA POSING FOR SUCCESS Sun's New Peer-to-Peer Technology Project. Available at: <http://java.sun.com/features/jxta.html>, April 2001
- 4 Sun Microsystems. Project JXTA FAQ. Available at: <http://www.jxta.org/project/www/docs/DomainFAQ.html>
- 5 Sun Microsystems. Project JXTA: Complete Access to an Expanded Web. Available at: <http://www.sun.com/software/jxta/go-live.html>
- 6 Sun Microsystems. JXTA: Helping To Build The X Internet. Available at: <http://www.sun.com/software/jxta/jxta-build.pdf>, JANUARY 23, 2002