

# 一种适用于动态/移动环境的新的组播协议

范 锐,程时端

(北京邮电大学国家重点实验室,北京 100876)

**摘 要:** 组播协议由于能够有效的利用网络带宽并降低服务器的负担因而受到广泛的关注.其中的共享树组播协议由于具有良好的扩展性而得到更多的支持,然而目前的共享树组播协议在动态移动环境中的性能很差,一些改进协议(单核移动协议)虽然可以很好的提高原有协议的性能但是所需要的额外开销很大.针对这种情况,本文提出了一种新的共享树组播协议-动态多核协议,该协议能够根据组播成员的分布动态的渐进的调整组播树的形状,从而能够在低开销的情况下为移动用户提供良好的性能.计算机仿真和数学分析都表明动态多核协议要明显优于共享树组播协议和单核移动协议.

**关键词:** IP 组播;组播协议;动态组播;移动组播

**中图分类号:** TN915.05 **文献标识码:** A **文章编号:** 0372-2112 (2002) 12A-1957-05

## A New Multicast Protocol for Dynamic/Mobile Environment

FAN Rui, CHENG Shi-duan

(National Laboratory, Beijing University of Posts and Telecommunications, Beijing 100876, China)

**Abstract:** Multicasting has gained much attention in recent years because it can efficiently utilize network bandwidth and reduce the load at application servers. The shared tree multicast protocol is more preferable due to its good scalability. At the same time, more and more mobile users require to participate in multicast communication. However, current shared tree multicast protocol performs poorly in dynamic/mobile environment. Some proposed protocols-one core migration protocol, improves the performance of shared tree protocol at the expense of large overhead in tree adjustment. This paper presents a new shared tree multicast protocol-dynamic multi-core protocol. This protocol can adjust the multicast tree according to the locations of multicast group membership gradually and thus provide good performance to group members with low cost. Computer simulation and mathematical analysis both show that dynamic multi-core protocol outperforms old shared tree protocol and one core migration protocol.

**Key words:** IP multicast; multicast protocol; dynamic multicast; mobile multicast

### 1 引言

随着计算机与网络技术的飞速发展,越来越多的应用需要利用组播协议,同时向多个目的地传送信息以提高网络资源利用率并减轻了服务器的负担,比如电视会议、远程教学等.到目前为止,固定网络下的组播通信协议已经得到了很大的发展.一类广泛采用的组播协议称作共享树协议.共享树协议分为 CBT(Core Based Tree)<sup>[1]</sup>协议和 PIM-SM(Protocol Independent Multicast-Sparse Mode)<sup>[2]</sup>协议.无论是 CBT 协议还是 PIM-SM 协议在创建组播树之前需要指定网络中的一个节点作为共享树的核心.其中 CBT 协议创建双向共享树,在组播树上的任意一个节点都可以直接把数据发送到组播树上;而 PIM-SM 协议创建的是单向共享树,组播组中的任何一个节点要想把数据发送到组播树上,都需要首先把数据发送到共享树的核上,再由核转发给所有组播成员.虽然 CBT 协议相对

于 PIM-SM 协议,具有较短的平均传送时延,不易引起业务流在核心的汇聚,但 CBT 协议无法有效解决对发送源的访问控制问题.要防止非法用户向组播组发送恶意数据,每个节点都需要装备策略信息,具备访问控制能力,而这显然是不可行的<sup>[3]</sup>.因而 PIM-SM 协议得到了更为广泛的支持.

PIM-SM 协议在组播树建立之前,要在网络中选择一点作为组播组的核心(也称作聚集点).所有要参加组播组的成员通过向该核心发送加入消息而加入到组播组中,并最终形成组播树.组播组所选择的核心将影响到最终形成的组播树的形状,乃至组播树的性能.因此核心的选择对于 PIM-SM 协议的性能起着非常重要的影响.

在固定网络中,当所有组播成员都能够事先确定位置时,可以通过一些启发式算法<sup>[4]</sup>来选择一个好的核心,从而得到优化的组播树.当组播成员的位置动态变化时,要选择合适的一个核心变得非常困难.文献<sup>[5]</sup>表明当组播成员位置动态变

化时,使用一个在通信开始前确定核心的组播树,其性能非常的差.换句话说,PIM-SM 协议虽然能够在固定网络下工作得很好,但是却不适合于动态/移动环境.

随着移动计算的日益普及,大量的组播成员可能是移动用户,因此需要为移动环境设计新的组播协议.文献[5]提出了一种改进 PIM-SM 协议的方案.该文献的基本思想是让 PIM-SM 协议中的聚集点在组播通信过程中动态迁移.在组播通信过程中,如果发现当前组播树的性能劣化,而根据另外一个被选核所生成的组播树的性能要优于当前组播树时,所有组播成员离开原组播树而加入到新核构成的组播树,从而改善组播树的性能.文献[5]的缺点在于为了得到优化的性能,需要重建整个组播树,开销比较大,而且由于组播树重建的时延比较长,在此期间容易导致大量数据包的丢失.文献[6]和文献[7]提出了类似的算法,但是都需要重建整个组播树以改善组播树的性能.本文提出了一种新的组播协议-动态多核共享树.该协议通过采用多个动态变动的核能够根据组用户的位置变化,动态渐进的调整组播树的形状,从而改善组播树的性能,而由组播树调整所引起的开销以及数据包的丢失都很少,从而很好的解决了目前组播协议在动态/移动环境下所面临的问题.同时该协议还解决了 PIM-SM 协议和 CBT 协议的缺点,既能够很好的解决发送源接纳控制的问题,又不会造成业务流在核心的汇聚.

## 2 动态多核共享树协议

从引言中可以看到,目前还没有一个适合于动态/移动环境下的组播协议.传统的 PIM-SM 由于核心的位置固定,无法适应组播成员动态变化的情况,导致性能很差.文献[5]中所提出的单核移动协议,虽然可以提高组播树的性能,但需要重建整个组播树,开销很大;而且由于重建组播树的时间比较长,在此期间会有大量数据包的丢失,因此也不适用于动态/移动环境.我们希望得到一个新的组播协议既能够改变组播树的形状以适应组成员位置动态变化的需求同时又尽量减少由此所导致的开销.由于组播成员数量众多,且行为各异,某些组成员位置变化导致自身性能的劣化并不意味着其他用户的性能也变坏了,因此在对组播树作调整时,没有必要作全局优化重建整个组播树,而可以修改组播树做局部优化.组成员可以根据自己所得到的性能来决定是否需要重新建立组播

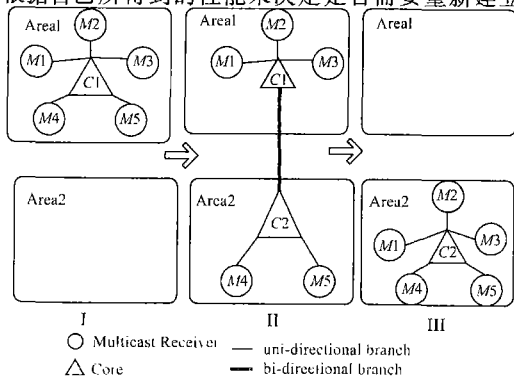


图 1 动态多核协议的基本思想

树.性能较差的组成员可以在局部重建组播树以改善它们得到的性能,而性能还可满足要求的组播成员可以保持该局部组播树不变.根据这一思想,在该协议中,组播树的核不再是一个固定点也不是一个动态移动的点,而是几个动态变化的点.它们根据组成员的分布,动态的增加和删除,这就是动态多核共享树的概念.通过图 1 来说明此方案(为了更好的说明动态多核协议的思想,图 1 中省略了连接组播用户与核心的子网组播路由器,并且假定任何一个主机的移动都会导致组播树的改变.).

组播通信开始前,所有组播成员都位于区域 I,当前核为 C1.在组播通信过程中,组成员 M4, M5 离开区域 I,到达区域 II,(假设区域 I 与区域 II 距离较远),这时 C1 就不再适合于 M4 和 M5,我们在区域 II 为 M4 和 M5 选择一个新核 C2,同时保留 C1,局部变动原来的组播树,改善 M4 和 M5 的性能.当组成员 M1, M2, M3 也移动到区域 II 时,此时区域 I 中已经没有组播用户,则把 C1 删除掉以减少需要维护的核的数量.通过这种方式既改善了组播树,提高了组播成员性能,又减少了由于重建整个组播树所需要的大量的开销以及在此期间数据包的丢失.

动态多核协议保留了 PIM-SM 协议对业务源作接纳控制的能力,即任何一个数据源在向组播组发送数据之前,需要首先把数据发送到核心.但当组播组中有多个核,且以这些核为根的多个子树之间相互独立或通过单向链路相连(类似于 PIM-SM)时,要使每个组成员都接收到组播数据,数据源必须向每个核发送数据,这会导致较大的开销.为了解决这个问题,我们在核之间构造一棵类似于 CBT 方式的双向树,而核与组播成员之间还是通过单向树相连.这样数据源只要向任意一个核发送数据,所有组播成员都可以接收到.使用这种方式,既保持了 PIM-SM 易于做接纳控制的优点,同时又避免了上面提到的数据源要向多个核发送数据,造成开销过大的问题.

## 3 动态多核协议的几个技术问题

### 3.1 核心的选择

动态多核协议虽然能够根据组成员位置的变化,动态的增加和删除核,但是能够充当核的节点的数量是有限的,位置是预先确定的,而不是任意节点都可以作为核,否则会失去接纳访问控制的优点,同时还会造成大量的开销.数据源或组播接收者可以通过两种方法获得当前组中核的位置,一种是在组播通信前静态配置,另外一种是通过 anycast 协议动态获得.动态多核协议采用后一种方案.

核的位置对组播树的性能有重要的影响.根据文献[5]的研究,核的选择策略应该是每一个转发域(transit domain)中需要一个核,那些具有潜在大量发送者或接收者的根域(stub domain)需要一个核,如果转发域的数量比较多,那么可以随机在这些转发域中选择核.

### 3.2 新核的生成

每个接收者根据它当前从组播组中得到的性能独立决定是否应该调整自己的组播树.如果它所得到的性能太差,超过

了门限值,那么它就向最近的被选核发送加入消息,请求加入以此点为核的树,但此时并不从原树中离开.被选核的位置可通过 anycast 协议得到.被选核收到来自某些组成员的加入请求后,与事先设置的门限作比较,如果该区域中有足够多的节点都要求调整它们的组播树,或者某些组播成员目前得到的性能很差,那么该核把自身设置为激活状态,并向那些发送了加入请求的组播成员返回加入确认信息,同时利用双向链路把自己连接到距自己最近的在树上的激活核.

组成员发出的加入消息都有时间限制,如果某个组成员在发送加入消息后的某个时限内没有收到确认信息,表明自身不足以启动一个新核的建立,它仍然停留在原树上,否则它就向原核发送离开消息,从新核接收数据从而改善自身的性能.需要强调的是,虽然每个接收者可以独立做出是否调整组播树的决定,但它的决定需要由被选核认可方启动一个新核的建立.

### 3.3 核的调整

所有激活核(Active Core)构成一棵双向树.由于组成员的变化,某些激活核可能需要离开组播树,因此需要对该树作调整.我们的方案如下.每个激活核定期检查目前是否有组成员通过它连接到组播树上以及是否有数据源通过它向组中其他用户转发数据.当这两个条件都不成立时,该核可能需要离开组播树,此时该核把状态设置为半激活态(Half Active),为了防止状态在短期内反复变化造成核不断的加入或离开树,只有当上述两条件同时不成立的时间大于某个门限后,才将该核的状态标记为半激活.在树上的每个核定期把自身当前状态发送到网络上.状态信息遇到在树上的其他核时停止转发.通过这种方式,每个在树上的核都可以发现自己的"邻居"核.当某个激活核发现组播树上出现半激活核时,开始带优先级的深度遍历.遍历过程中优先考虑激活核,只有所有激活邻居核都遍历完毕,才遍历半激活的核.通过遍历可发现半激活的核是否需要留在树上(如果半激活的核处于两个激活核的最短路径上,则该核不应该离开组播树),从而能够删除不需要的核,达到组播树优化的目的.从任何一个激活核都可以遍

历,为了防止所有激活核同时开始遍历造成不必要的开销,只有拥有令牌(token)的核才可以启动遍历过程.如果拥有令牌的核变为半激活状态,那么它需要把令牌传递给激活状态的核.我们用图 2 来说明如何对核构成的树作调整.

核调整的步骤

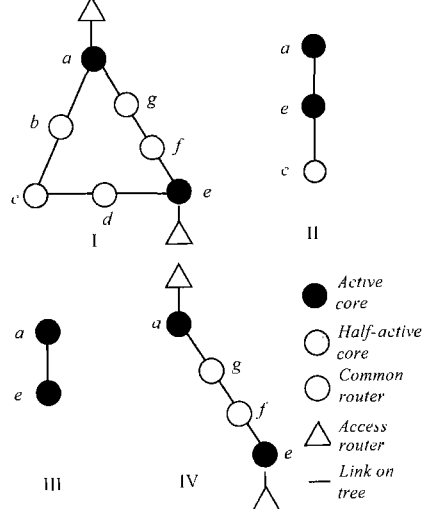


图 2 双向树的调整

骤如下:

- (1)组播树上的每个核通过交换状态信息得知它的邻居核的状态(激活核,半激活核等);
- (2)拥有令牌的核执行有优先级的深度遍历(首先遍历激活的邻居核,若有两个激活的邻居核,首先遍历最近的邻居核);
- (3)处于末端的非激活核离开树;
- (4)激活核重组得到优化的树.

### 3.4 数据的发送

每个数据源发送数据时可以通过 anycast 协议找到距离自己最近的核来发送数据.如果在组播通信过程中,距某个数据源最近的核改变了,可以由该核向数据源发送通知,使其随后通过此核转发数据.

### 3.5 动态多核协议的完整描述

综上所述,我们在下面完整地描述动态多核组播协议.

- (1)在组播通信开始前,为组播组选择合适的备选聚集点集合  $G_c$ ,所有的备选聚集点都处在一个 anycast 组中  $G_A$ ,选择其中任意一个作为当前的活动核  $C_i$ ,并给  $C_i$  赋予一个令牌 token,组成员获知  $C_i$  加入到以此核为根的共享树  $T$  中.数据源获知  $C_i$ ,向  $C_i$  发送单播封装的组播数据,开始组播通信过程.

- (2)每个组播接收者  $m$  监测它所得到的性能  $h_c$ ,如果  $h_c > \alpha h_0$  ( $\alpha$  是一个可设置的常量参数,  $h_0$  是性能门限),则启动树的重建过程 2,否则仍在 1 中.

- (3) $m$  利用 anycast 协议找出距自己最近的被选核  $C_s$ ,如果  $C_s \neq C_i$ ,向  $C_s$  发送加入消息,其中包含有  $\beta = h_c/h_0$ ,表示它性能恶化的程度,以及目前的数据源集合  $S$ ,并启动计时器  $t_a$ .  $C_s$  收到来自  $m$  的请求后,记录向它发送重建请求的组成员的数量  $n$ ,如果  $n > n_0$  或者  $\beta > \beta_0$  ( $n_0$  和  $\beta_0$  为门限值)表明它附近的组成员性能足够差,则把自身标记为激活状态,利用 anycast 协议找到距自己最近的激活核,向它发送 join 消息,构建双向链路,同时向发送确认消息,通知在集合  $S$  中的数据源有新的核加入到组播树中,否则仍然停留在非激活态.在  $t_a$  超时之前,如果  $m$  收到了来自  $C_s$  的确认消息,它向  $C_i$  发送 leave 消息,否则仍然停留在原树上.

- (4)每个在树上的核  $C_i$  监测自身的状态,并定期向网络发送状态信息.如果在  $t' > \tau$  的时间段内,没有数据源通过它向组用户转发数据也没有组成员通过它连接到组播树上,它的状态为半激活  $H$ ,否则为激活  $A$ .如果  $H$  状态的核拥有 token,则把 token 传递给  $A$  状态的核.

- (5)每  $t > t'$  时,拥有 token 的核发起带有优先级的深度优先遍历,并根据遍历结果调整由激活核和半激活核构成的双向树.

- (6)数据源在发送数据期间,如果收到来自某个核的通知消息,则重新使用 anycast 协议,以便找出目前最近的激活核,向它发送数据.

## 4 协议的性能分析与仿真

### 4.1 仿真模型

我们利用 ns-2.1b9 仿真动态多核协议的性能.仿真模型

与其他文献中类似<sup>[8]</sup>. 仿真中所用到的拓扑图按照文献[8]中的方法随机生成, 仿真拓扑图类似于真实 Internet 网络的情况, 分为转发域和根域. 最终生成的随机拓扑图如图 3 所示.

为模仿组成员动态变化的情况, 假设每个用户随机的加入或离开组, 用户到达或离开组的事件服从泊松分布, 即每个用户在组中停留时间服从指数分布. 用户加入还是离开组由随机模型确定, 该随机模型用公式表示为

$$P_c(m) = \frac{\gamma(N-m)}{\gamma(N-m) + (1-\gamma)m} \quad (1)$$

其中:  $m$  表示当前在组播组中的用户数;  $N$  为所有的终端用户;  $\gamma$  是一个在  $(0-1)$  之间的实数. 参数  $\gamma$  决定了处于平衡状

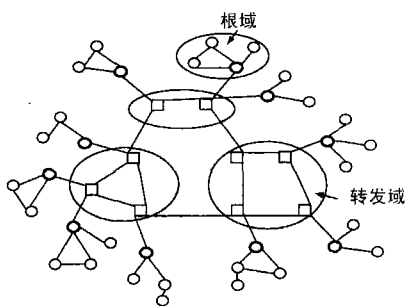


图 3 仿真所用到的拓扑结构图

态时, 组中用户的数量. 用户加入或离开组的条件是当  $P_c(m) > 1/2$  加入组, 否则离开组. 如果要求用户加入组, 随机选择一个还没有加入组的用户, 加入到组中; 如果要求用户离开组, 随机选择一个在组中的用户, 让它离开组.

仿真的目的在于比较动态多核协议与 PIM-SM 协议以及单核移动协议的性能. 比较的性能参数包括协议所生成的组播树的平均大小, 组播用户从数据源接收数据的平均时间, 以及在组播树调整期间, 组播用户丢失数据包的平均数量. 仿真中的参数设置如下, 每次生成的随机拓扑图包含有 3 个转发域和 10 个根域, 总的用户的数量为 50,  $\gamma$  取值为 0.4, 仿真结果是 20 次随机结果的平均值.

### 4.2 仿真结果

首先比较动态多核协议与 PIM-SM 协议用户接收数据的平均时间, 仿真结果示于图 4. 从图 4 中可以看到, 除了在仿真开始时, 动态多核协议与 PIM-SM 协议的平均时延相同外, 在其他时刻, 动态多核协议的平均时延都小于 PIM-SM 协议, 前者只有后者的 70% 左右. 这表明动态多核协议的组播用户能够以更短的时延接收数据源发来的组播包.

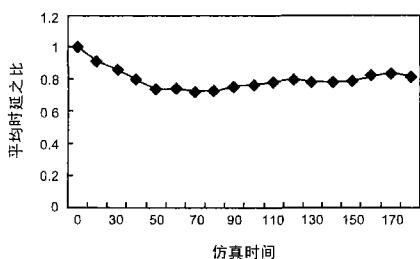


图 4 动态多核与 PIM-SM 组播用户接收数据的平均时延之比

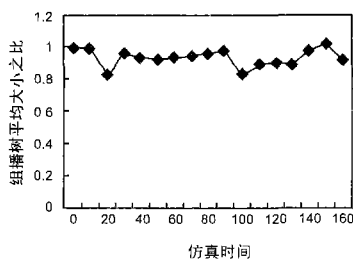


图 5 动态多核协议与 PIM-SM 协议组播树平均大小之比

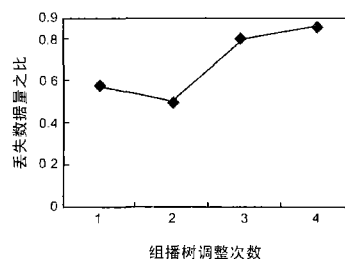


图 6 动态多核协议与单核移动协议数据包丢失数之比

下面, 比较动态多核协议与 PIM-SM 协议所建立的组播树的大小, 仿真结果示于图 5. 从图 5 中可以看出, 除了极个别的情况下, 动态多核协议创建的组播树, 都要小于 PIM-SM 协议所创建的组播树, 平均而言, 前者只有后者的 90% 左右. 在个别时刻, 动态多核协议所创建的组播树比 PIM-SM 协议所创建的组播树略大, 这是由于动态多核协议未能及时把多余核去掉造成的.

以下是动态多核协议与单核移动协议的性能对比, 我们重点考察由于组播树重建而引起的数据包丢失的情况. 为了能够更方便的对比这两种协议, 所设定的组播树调整的条件是, 当任意一个组播成员所接收数据包的时延超过一个最大门限时就调整组播树. 收集了每个协议组播树分别调整 4 次时, 数据包的丢失情况, 如图 6 所示.

从图中可以看到, 每一次组播树调整, 由动态多核协议引起的数据包的丢失都要比单核移动协议少, 这是因为动态多核协议并不是重建整个组播树, 而是在局部做调整, 因此那些没有受到影响的组播接收者不会丢失它们的数据.

### 4.3 性能分析

利用数学分析比较动态多核协议与单核移动协议调整组播树所需的开销. 为了简化分析, 假定组播树的改变仅由组播成员位置的变迁引起, 也就是组播组中只有移动用户, 而且它

们直到组播应用结束才离开组播组. 网络中移动用户的总数为  $P$ , 它们均匀分布在  $N$  个子网当中并以相同的速率自由移动, 稳态下平均每个子网中的移动用户数为  $M = P/N$ , 由于每个移动主机可以独立的进入或离开一个子网, 移动主机到达或离开子网的过程可看作是泊松过程, 因此可以用  $M/M/\infty/M$  排队模型来分析每个子网中的移动用户数<sup>[9]</sup>. 设移动用户到达子网的泊松强度为  $\lambda$ , 在子网中停留的时间为  $1/\mu$ , 一个子网中有  $k$  个用户的概率  $p_k$  可以由下式得到

$$p_k = \begin{cases} \frac{\left(\frac{\lambda}{\mu}\right)^k \binom{M}{k}}{\left(1 + \lambda/\mu\right)^M}, & 0 \leq k \leq M \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

当子网中有一个以上的组播用户时, 该子网在组播树上, 因此由式(2), 一个子网在组播树上的概率为

$$\sum_{k=1}^M p_k = 1 - p_0 = 1 - \frac{1}{\left(1 + \lambda/\mu\right)^M} \quad (3)$$

而当子网中没有移动用户时, 该子网不在组播树上, 相应的概率为  $p_0 = \frac{1}{\left(1 + \lambda/\mu\right)^M}$ . 因此  $N$  个子网中在组播树上的子网个数的期望值为

$$G = N \left[ 1 - \frac{1}{\left(1 + \lambda/\mu\right)^M} \right] \quad (4)$$

相应的  $N$  个子网中,不在组播树上的子网数量为

$$G_0 = N * p_0 = \frac{N}{(1 + \lambda/\mu)^M} \quad (5)$$

而在组播树上且只有一个移动用户的子网数量为

$$G_1 = N * p_1 = \frac{NM\lambda}{\mu(1 + \lambda/\mu)^M} \quad (6)$$

$G_0$  是那些可能会新加入到组播树上的子网数量的上限,而  $G_1$  则是那些可能会离开组播树的子网数量的上限,换句话说,  $G_0 + G_1$  是在动态多核协议中,组播树作局部调整的子网数量的上限;与此对应,  $G$  则是在单核移动协议中,组播树调整的子网数量的下限。由于调整组播树的开销与涉及到的子网数量成比例<sup>[10]</sup>,因此可以用  $G$  表示单核移动协议组播树调整的开销,  $G_0 + G_1$  表示动态多核协议组播树调整的开销。给各个变量赋值后,我们得到两种方案开销之比示于图 7。从图中可以看到,动态多核协议与单核移动协议相比,能够有效地减少组播树调整所引起的开销,而且随着子网中平均用户数的增加或者子网利用率的提高,所得到的增益也越大。

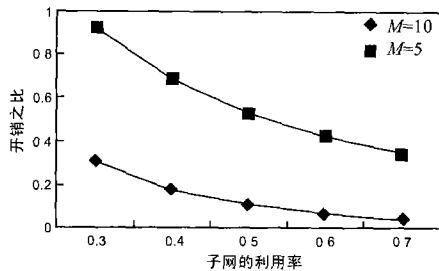


图 7 动态多核协议与单核移动协议组播树调整开销之比

综合计算机仿真与数学分析的结果不难看出,动态多核协议与 PIM-SM 协议相比能够有效地减少组播用户接收数据的时延和组播树所占用的网络带宽,而与单核移动协议相比又能够减少组播树调整所需要的额外开销以及在此期间所造成的数据包丢失,因此动态多核协议更适用于动态/移动环境下的组播。

## 5 结论

组播协议由于能够有效的利用网络带宽并降低服务器的负担因而受到广泛的关注。而其中的共享树组播协议由于它所具有的良好扩展性而得到更多的支持。与此同时在动态移动环境中使用组播技术是未来网络发展的趋势,然而目前的共享树组播协议由于在设计时没有考虑动态移动用户,因而在动态移动环境中的性能很差。一些改进协议(单核移动协议)虽然可以很好的提高原有协议的性能但是所需要的额外开销很大。针对这种情况,本文提出了一种新的共享树组播协议—动态多核协议,该协议通过多核的使用能够根据组播成员的分布动态的渐进的调整组播树的形状,从而能够在低开销的情况下为移动用户提供良好的性能。计算机仿真和数学

分析都表明该协议要明显优于共享树组播协议和单核移动协议。

## 参考文献:

- [1] T Ballardie. Core Based Trees Multicast Routing[S]. RFC 2189, Sept. 1997.
- [2] D Estrin, et al. Protocol Independent Multicast-Sparse Mode Protocol Specification[S]. RFC 2362, Oct. 1999.
- [3] 马钰璐.多核单向共享树多播路由协议[J].计算机学报,2001,24(7):710-715.
- [4] Hwa-chun Lin, Shou-chuan Lai. Core placement for the core based tree multicast routing architecture[A]. Global Telecommunications Conference[C]. Sydney, Australia, 1998. II 1049-1053.
- [5] M J Donahoo, E W Zegura. Core migration for dynamic multicast routing[A]. Fifth International Conference on Computer Communications and Networks, ICCCN96[C]. Washington DC, USA. 1996. 92-98.
- [6] Raghavendra, A D Rai, S. Multicast routing in internetworks using dynamic core based trees[A]. Proceedings of the 1996 IEEE Fifteenth Annual International Phoenix Conference on Computers and Communications[C]. Scottsdale, USA, 1996. 232-238.
- [7] Koubaa, H Fleury, E. Active multicast core migration[A]. Proceedings. IEEE International Conference on Networks, ICON 2000[C]. Singapore, 2000. 346-350.
- [8] H C Lin, S C Lai, VTDM-A dynamic multicast routing algorithm[A]. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings of INFOCOM'98[C]. San Francisco, USA, 1998. III 1426-1432.
- [9] Banerjee N Das S K. Analysis of mobile multicasting in IP-based wireless cellular networks[A]. IEEE International Conference on Communications, 2002. ICC 2002[C]. NY, 2002. V 3388-3392.
- [10] J Chuang, M Sirbu. Pricing multicast communication: A cost-based approach[A]. Proceeding of the 1998 INET[C]. Geneva Switzerland, 1998. 281-297.

## 作者简介:



范 锐 男,1975 年 4 月生于陕西省西安市,1997 年毕业于北京邮电大学计算机学院,获硕士学位。现为北京邮电大学程控交换技术与通信网国家实验室博士生。研究方向:网络流量控制与拥塞控制,移动 IP 技术,组播技术。

程时端 女,1946 年 6 月生于上海,北京邮电大学程控交换技术与通信网国家实验室教授,通信与信息系统学科博士生导师。研究内容包括:IP 网络流量控制与拥塞控制,移动 IP 技术,组播技术,路由器队列调度与缓存管理技术等。