

# 基于域饱和度的二维应用层多播模型

许建真<sup>1,2</sup> 全宗勇<sup>2</sup> 张福炎<sup>1</sup>

(南京大学计算机软件新技术国家重点实验室 南京 210093)<sup>1</sup>

(南京邮电大学软件学院校园网中心 南京 210003)<sup>2</sup>

**摘要** 提出了一种基于域饱和度的二维应用层多播模型 TDRS,该模型以域为基础,具有控制开销小、高效、分布式的结构。重点阐述了多播节点的加入和退出过程,首先以域饱和度为基础平衡选取目标域,接着以 DCMR 算法二次择优产生域内路径,最终达到 ALM 拓扑结构的整体优化。仿真结果表明 TDRS 模型能有效减少网络传输冗余,混乱情况下自愈性强、传输数据速度快,具有低延迟、低链路压力的特点,同时解决了现实网络中部分节点服务能力限制的问题,增强了系统的实用性和扩展性。

**关键词** 应用层多播,域饱和度,二维,度约束

**中图分类号** TP393 **文献标识码** A

## Two-dimensional Model Based on Region Saturation for Application Layer Multicast

XU Jian-zhen<sup>1,2</sup> QUAN Zong-yong<sup>2</sup> ZHANG Fu-yan<sup>1</sup>

(State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China)<sup>1</sup>

(Campus Computer Network Center, College of Software, Nanjing University of Posts & Telecommunications, Nanjing 210003, China)<sup>2</sup>

**Abstract** This paper proposed a two-dimensional model based on region saturation for ALM. The model has many merits, such as little control overhead, high efficiency, distributed structure. This paper focused on nodes joining and departure procedures, so region saturation is defined as the standard for the elementary selection, and DCMR algorithm for further selection, this achieves an entire optimization of ALM topology. The simulation results show that the model is an efficient mechanism to reduce traffic redundancy in the network, transmit data with high speed and resume itself in a disordered situation. It has a good match to the underlying physical network such that the link stress and the end-to-end delay can be effectively reduced. In the mean time, real-life network constraints can be handled.

**Keywords** Application layer multicast (ALM), Region saturation, Two dimension, Degree constraints

多播是一种实现一对多的通讯技术,可以广泛应用于大规模信息发布、实时视频流媒体会议、分布交互式应用等服务。IP 多播是一种较早出现的多播实现机制,它在网络层实现多播技术,虽然具有较高的传输效率,但由于各种技术和非技术的原因,并没有取得很好的发展。随后发展而来的应用层多播技术 ALM (Application Layer Multicast),与网络层多播依靠路由器复制数据不同,它不改变网络底层结构和内部组织,由端节点实施向前广播的功能。由于拥有较好的传输特性,又可以摆脱底层基础设施网络的限制而易于实现,应用层多播成为备受瞩目的多播实现机制。

目前已经出现了许多较完善的应用层多播模型,如 Yoid<sup>[1]</sup>, Narada<sup>[2]</sup>, NICE<sup>[3]</sup>等。无论采用何种机制,应用层多播的目的始终是以高效率、低代价让参与者享受多播服务。本文提出了一种基于域饱和度的二维应用层多播模型 TDRS (Two-Dimensional Model Based on Region Saturation),建立和维护网络的逻辑层和域内树结构,通过层来管理区域节点集,通过带度约束的最小半径支撑树算法 DCMR (Degree-

Constrained, Minimum Radius Spanning Tree Algorithm)来形成域内数据路径。

## 1 TDRS 模型设计

### 1.1 模型结构

现有的一些应用层多播存在着链路压力过大、效率低下等诸多不足。如网状优先的 Narada 模型,侧重于解决终端节点的服务带宽限制和源节点到用户节点的延迟等问题,并没有有效降低网络的连接压力;层次结构的 NICE 模型,在保证网络节点的邻近特性的同时,需要通过其它辅助策略来准确地选择具有高性能的父节点,耗费了大量的系统资源。

本节将讨论一种基于域饱和度的二维应用层多播模型 TDRS,模型组织结构详细描述如下:

(1)以域(Region)为基本逻辑管理单位,构造和维护一个基于域的多层(Layer)网络逻辑结构,网络由  $m(m \geq 2)$  层构成,层表示为  $L_i$ ,底层为  $L_0$ ,依次往上逐层递加;

(2)所有节点初始加入到  $L_0$  层,每层由  $l(l=1, 2, \dots, l)$

到稿日期:2008-01-14

许建真(1966-),男,副教授,主要研究方向为计算机通信网与信息安全等,E-mail:courage@njupt.edu.cn;全宗勇(1983-),男,硕士研究生,研究方向为计算机通信网与网间互连;张福炎(1938-),男,教授,博士生导师,主要研究方向为计算机应用技术等。

个域构成,每个域最多包含  $k$  个( $k$  为常数)节点,网络中的任一节点必属于一个特定的域;

(3)每个域有一个中心节点(Leader),中心节点由域周期性运行轮举算法选择综合性能最优的节点担任;

(4)任一节点在任何层只能存在于一个域,但中心节点同属于相邻层中不同的两个域;

(5)如果一节点处于  $L_i$  层的某个域中,它一定是  $L_0, \dots, L_{i-1}$  层的某个域中的中心节点;

(6) $N$  个节点规模的网络最多有  $\log_k N$  层,路径搜索采用分布式向前询问方式完成;

(7)从第 1 层开始,第  $i$  层的节点由第  $i-1$  层的中心节点组成,依此类推。同一层各域的中心节点相互连接形成环状,如图 1 所示, $L_0$  层存在 4 个域,A,B,C,D 分别为各域的中心节点,相互连接形成环 A-B-C-D。

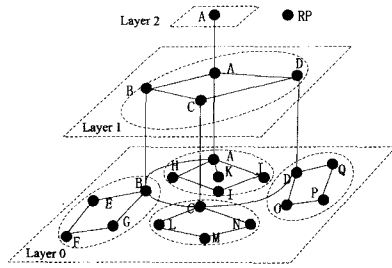


图 1 TDRS 多播模型组织结构

从图 1 可以看出,物理上节点位于第 0 层不同的域中,中心节点同属于两个相邻的层和不同域,并保存有相邻域中心节点的信息,通过映射函数产生域集合。模型结构中各成员的定义如下。

**普通节点(Host):**综合性能较弱,同域节点的性能和属性基本相似,构成一个资源和计算中间结果全面共享的拓扑网络。节点请求加入或离开网络时,通过向网络发布消息,建立或撤销连接,更新资源和服务目录,节点通过中心节点实现跨域访问。

**中心节点(Leader):**由域内综合性能最优的节点充当,逻辑上位于域的中心,是域间接口,负责域内网络节点管理,完成对等点间“握手”、工作结果备份和归档。中心节点维护一个域间和一个域内活动对等点的信息列表,新加入节点须向中心节点注册包括节点编号和地址信息。此外,中心节点还负责收集和反馈域内外的状态信息,及时处理节点的加入和离去,更新服务和网络状态信息。

**汇聚节点 PR(Rendezvous Point):**一般由具备较强处理能力的主机充当,逻辑上不属于任何多播组。PR 节点保存网络中每个域的中心节点的状态信息以及所提供的服务,负责引导新对等节点的加入,对整个网络进行管理和控制,但不负责数据包的传输和分发。

## 1.2 新成员加入

多播组的形成从数据源点的工作开始,希望得到多播服务的终端节点随后依据协议规则不断加入到多播组中,多播组逐渐扩大的过程即为新节点不断加入的过程。

任何新节点在加入多播组前先发送 Join\_Message 给 RP,然后选择最佳的父节点加入。RP 节点所存储的中心节点信息的初始化如表 1 所示,其中,

**Leader\_Num:**中心节点编号,其中包含了中心节点所在

的层编号,也包含了域编号,格式为 Layer\_Region,如  $L_0$  层的第一个域中心节点编号为 00\_01;

**IP:**中心节点的 IP 地址;

**Sat:**中心节点所在域的饱和度(Saturation),即域内节点的数量与域规模  $k$  之比,计算公式为  $Sat = n / k$ ;

**k:**中心节点所在域的规模,即域所允许容纳的最大成员数量;

**n:**域内当前存在节点的总数,由相应的 Leader 节点周期性更新。

**b:**定义为多播组中每条链路的带宽,在 TDRS 模型中单条链路带宽由 RP 来集中控制;

**L\_Time (s):**节点存在于多播组中的时间。

表 1 RP 所存储的中心节点初始化列表

Leader_Num	IP	Sat	k	b	n	L_Time
00_00	*.*.*.*	0	0	0	0	0.0

中心节点所存储的域内普通节点信息的初始化如表 2 所示,其中,

**Host\_Num:**普通节点编号,如  $L_0$  层第一个域第一个节点编号为 00\_01\_0001;

**IP:**普通节点的 IP 地址;

**Out\_Degree:**节点的实际出度,即节点当前所服务的子节点的数目;

**BW:**节点所拥有的接入带宽;

**Src (kb):**节点所持有数据资源量大小;

**H\_Time (s):**节点存在于多播组中的时间。

表 2 中心节点存储的域内节点初始化列表

Host_Num	IP	Out_Degree	BW	Src	H_Time
00_00_0000	*.*.*.*	0	0	0	0.0

已存在于多播组的节点会以  $\Delta T_1$  的间隔时间周期性地向所在域的 Leader 节点发送 Update 更新报文,更新其在 Leader 节点信息表中的 Out\_Degree, BW 和 L\_Time 等值。而 Leader 节点也会以  $\Delta T_0$  的间隔时间周期性地向 RP 节点发送 Update 更新报文,更新其在 RP 信息表中的各项参数值。

假设当前多播组内有  $N$  个节点,则可用集合  $G = \{1, 2, 3, \dots, N-1, N\}$  表示多播组, $N+1$  节点(记为 X)到达并试图加入多播组,加入过程描述如下:

(1)新节点 X 首先向 RP 发送 Join\_Message 报文,如图 2 所示。

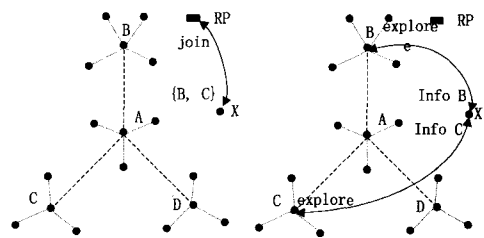


图 2 新成员加入示意图

(2) RP 在收到 X 节点的多播组加入请求后,检索所存储的中心节点信息表。如果此时信息表中各中心节点的 Sat 值

不全为 1,那么将表项按 Sat 值从小到大升序排列,选取前  $M$  ( $M$  为常数)个中心节点所在域作为目标加入域集,并将  $IP, n$  和  $L\_Time$  值反馈给  $X$  节点,同时 RP 发送给  $X$  节点的数据信息也包含多播组总的运行时间  $T$  以及后续工作所需要的运算规则。

(3) 如果此时 RP 信息表中各项的 Sat 值均为 1,说明此时各域已经饱和,则增加一个新域,任命  $X$  节点充当该域临时中心节点,同时更新 RP 中的数据表项。

(4)  $X$  节点收到 RP 节点反馈的数据信息后,首先向这些中心节点发送 Explore 数据包,获取与这些中心节点间的跳数(hop)以及相关参数值,并按照式(1)进行运算。其中定义  $D_i$  为量化的网络路径参考值,下标  $i$  表示对应为  $Leader\_Num = i$  的节点,  $H_i$  表示节点到  $X$  的跳数,  $Src_i$  为中心节点所在域内的各成员拥有的数据资源量,  $S$  为资源总量,  $T$  为系统存在总时间,其中选取平衡因子  $0 \leq \alpha, \beta, \gamma \leq 1$  满足  $\alpha + \beta + \gamma = 1$ :

$$D_i = \frac{\alpha}{H_i} + \frac{\beta * \sum_{i=1}^n Src_i}{S} + \frac{\gamma * L\_Time_i}{T} \quad (1)$$

(5) 选取  $D_i$  值最大的那个节点作为  $X$  节点所要加入域的中心节点。同时选择  $D_i$  值次小的节点作为冗余备份节点。

确定加入的域后,  $X$  节点还必须选择域内的某个节点作为父节点,该问题可视域内子树构建问题。

**定义** 图  $G=(V, E), |V|=n, \forall v \in V, \forall e \in E, c(e) \in Z$ , 每个节点  $v$  最大出度为  $d_{\max}[v] = \text{INT}(BW/b)$ , 即节点所拥有的接入带宽与单条链路带宽比值取整部分。寻找图  $G$  的支撑树  $T$ , 使得  $\forall v \in V, d_T(v) \leq d_{\max}[v]$ , 并且树  $T$  的半径最小。

**定理 1** 图  $G=(V, E)$  是一个带权连通图,  $U$  是顶点集合  $V$  的一个非空子集。若  $(u, v)$  是一条具有最小权值的边, 其中,  $u \in U, v \in V-U$ , 则必存在一棵包含边  $(u, v)$  的最小半径生成树。

**证明:** 可以用反证法加以证明。假设图  $G$  的任何一棵最小半径生成树都不包括  $(u, v)$ , 设  $T$  是  $G$  的一棵最小半径生成树。当将  $(u, v)$  加到  $T$  中时,  $T$  中必定存在一条包含边  $(u, v)$  的回路。另一方面, 由于  $T$  是生成树, 则  $T$  上必定存在另一条边  $(u', v')$ , 其中  $u' \in U, v' \in V-U$ , 且  $u$  和  $u'$  之间、 $v$  和  $v'$  之间均有路径相通。删除边  $(u', v')$  便可消除回路, 并可同时得到另一棵生成树  $T'$ 。因为边  $(u, v)$  的长度不超过  $(u', v')$ , 则  $T'$  的半径亦不超过  $T$ ,  $T'$  包含  $(u, v)$ , 故与假设相矛盾, 定理 1 得证。

本文以 Prim<sup>[4]</sup> 算法为基础, 同时结合 Kruskal<sup>[5]</sup> 算法的特点, 提出带度约束的最小半径支撑树算法 DCMR (Degree-Constrained, Minimum Radius Spanning Tree Algorithm), 算法描述如下:

(1) 随机选择父节点  $E_1^T, E_2^T, F \leftarrow E_1^T E_2^T, rd(v) = d_{\max}[v], v \in V, E_T = \Phi$ ;

(2) 统计  $E_1^T, E_2^T$  中各节点出现的次数, 记为  $pd(v), v \in V$ ;

(3) 随机选择两个出度未达到  $d_{\max}[v]$  的节点  $v_1, v_2$ , 如果  $pd(v_1) < pd(v_2)$ , 令  $v_1 = v_2$ ;

(4) 从  $F$  中选择和  $v_1$  连接的边  $e_1(v_1, v_2), e_2(v_1, v_3)$ , 如果  $c(e_1) > c(e_2)$ , 令  $v_2 = v_3$ ;

(5) 如果  $rd(v_2) > 0$ , 并且加入边  $e(v_1, v_2)$  不会形成回路, 则  $E_T \leftarrow E_T \cup e(v_1, v_2), F = F - e(v_1, v_2)$ ;

(6) 找出两个不连通的分支  $u_1, u_2$ , 计算连通分支中节点  $v$  与其它节点的最大距离  $\delta(v), v \in u_1 \cup u_2$ ;

(7) 合并  $u_1, u_2$ , 使得  $\max(\delta(v))$  最小,  $v \in u_1 \cup u_2$ ;

(8)  $E_T \leftarrow E_T \cup e(u_1, u_2)$ , 重复(1)-(9), 直到  $|E_T| = |V| - 1$  结束。

DCMR 旨在构建出域内最小半径支撑树, 算法的思想是把切分后两个不连通的部分树通过一条链路连接, 使得形成的新树半径最小。该算法采用边集对支撑树进行定义, 使得节点的加入和删除操作变得简单, 同时引入了贪婪算法的策略, 极大提高了算法的收敛速度。

### 1.3 中心节点提升

假设  $N_{(i,j)}$  表示第  $i$  层第  $j$  个域的中心节点, TDRS 模型希望综合性能较强的中心节点处在拓扑的高层。然而, 由于网络状态的不断改变和新节点的加入使得系统的结构不是很合理, 因此采用一种动态规划的方法<sup>[6]</sup>。根据各节点的资源占有量、时延、带宽、抖动以及在系统中的存在时间, 采用加权平均的方法计算出节点的优先级。

设节点在系统中的存在时间为  $L\_Time$ , 存在期间接收的数据为  $D_r(N_{(i,j)})$ , 发送的数据为  $D_s(N_{(i,j)})$ , 则可得节点存在期间的数据流量总量为

$$D(N_{(i,j)}) = D_r(N_{(i,j)}) + D_s(N_{(i,j)}) \quad (2)$$

也可计算出  $N_{(i,j)}$  节点转发效率为

$$\rho = D_s(N_{(i,j)}) / D(N_{(i,j)}) = D_s(N_{(i,j)}) / (D_r(N_{(i,j)}) + D_s(N_{(i,j)})) \quad (3)$$

判断一个中心节点是否应该向上提升, 首先要考察该节点的转发效率。主要根据该节点的转发效率  $\rho$  来决定, 但光有效率还不够, 节点若想向其他节点传送数据, 还需要本身拥有大量数据, 所以可用节点的资源总量  $S(N_{(i,j)})$  作为第二个参数, 以决定节点上浮的层数。

再考虑节点在系统中的存在时间, 让在系统中长时间存在的节点充当上游的主干节点, 有利于提高系统的稳定性。根据节点抖动对系统的影响, 多次加入、退出系统后, 只考虑最后一次。

上述的参数都具有即时性, 则  $t = L\_Time$  时节点  $N_{(i,j)}$  的优先级为

$$P(N_{(i,j)}) = e^{\frac{\rho}{1-\rho}} + \beta \frac{S(N_{(i,j)})}{S} + \gamma \frac{t}{T} + \delta \frac{1}{H} \quad (4)$$

$(0 < \alpha, \beta, \gamma, \delta \leq 1)$

其中  $T$  为系统稳定运行的时间,  $H$  为中心节点到源节点的跳数 hop。将优先级划分与系统分层对应, 即一定优先级的节点应该属于同一层。整个系统稳定后的状况一定是上层节点的优先级比下层高, 同层之间则属于同一范围。据此在数据传输过程中, 通过 heart-beat 算法动态改变拓扑中各域中心节点的层次, 让拥有大量资源的节点向上游浮动。

### 1.4 成员失效处理

参与应用层多播的节点均为拥有自主能力的主机, 这些主机随时可能正常地退出多播服务或因意外异常地离开多播组, 这就要求 ALM 模型拥有很强的自愈能力。TDRS 模型对于成员的离开分为两种情况:

(1) 普通节点离开。这种情况最为常见, 如果处于叶子

位置的节点正常离开多播组,则该节点向其父节点以及所在域的中心节点发送离开消息,并由中心节点删除数据表中该节点项;如果叶子节点异常离开,经过连续多个  $\Delta T_1$  时间后中心节点还未收到该节点的 Update 更新报文,则中心节点认为该节点已离开并做出相应的删除处理。如果离开的节点为非叶子节点,即该离开节点存在子节点,那么节点正常离开时向其父节点和子节点以及中心节点发送离开消息。而非叶子节点异常离开依然由中心节点通过 Update 更新报文进行监控,连续多个  $\Delta T_1$  时间后还未收到该节点的更新报文,则中心节点认为该节点已离开,通知其子节点。子节点收到父节点离开的消息后,立刻查找加入该域时的预留加入节点,重新加入多播组。如果预留加入节点的出度已经达到  $d_{\max}[v]$ ,则子节点重新运行,加入算法。

(2) 中心节点离开。由于中心节点的特殊地位, TDRS 模型在构建多播组的时候,将处于同一个层次不同域的中心节点相互连接,形成环保护。当某个域的中心节点离开多播组时,域中的其它节点按照邻近原则,从环上寻找 IP 最近且出度未超过  $d_{\max}[v]$  的中心节点所在域加入。

TDRS 模型利用成员冗余信息的存储来解决节点离开的问题,用切实的数据预留将链路的选择具体化,使得模型拥有很强的自愈能力,高效地解决节点离开所带来的影响。

## 2 模型性能

一个好的应用层组播协议,除了要有好的网络资源利用率和传输效率之外,还应该要有适中的控制负载、良好的网络负载平衡,并且要能满足接收者的性能需求。目前还没有一个用于衡量应用层组播协议的效率和性能的统一标准,但常用的指标包括:

(1) 连接压力(Stress)。定义为网络拓扑上同一条物理链路上相同数据包的个数,它反映出底层带宽消耗和带宽利用率。对于网络层多播,由于没有冗余数据包复制,因此网络中的每条链路上的压力为 1。

(2) 伸展率(Stretch)。即成员多播路径长度与直接单播路径长度之比。伸展率也反映出节点的网络时延,相同网络拓扑上节点的伸展率越高,时延也越大。

设一个采用 TDRS 模型结构的拓扑网络,由 3 层  $L_0, L_1, L_2$  构成,网络规模为  $N$ ,域规模为  $k$ 。  $A_0$  为网络中位于  $L_0$  层某一域的普通节点,该域的中心节点为  $A, B$  为  $L_0$  层的另一域的中心节点;在  $L_1$  层,  $A, B$  属于同一域,且  $B$  是中心节点,可知  $B$  同时也位于  $L_2$  层。

设  $S$  为源点,从  $S$  到节点  $A_0$  直接单播路径长度为  $R, S$  到  $A_0$  的多播路径为  $S \rightarrow B \rightarrow A \rightarrow A_0$ ,多播路径长度用  $r$  表示。 $S$  到节点  $B$  的路径为  $S \rightarrow B$ ,长度为  $r_2$ ;  $B$  到节点  $A$  的路径为  $B \rightarrow A$ ,长度为  $r_1$ ;  $A$  到节点  $A_0$  的路径为  $A \rightarrow A_0$ ,长度为  $r_0$ ,则  $S$  到  $A_0$  应用层多播路径长度为:

$$r = r_0 + r_1 + r_2 = \sum_{i=0}^2 r_i \quad (5)$$

定义多播路径与单播路径的夹角为路径偏移度,用  $\beta$  表示,例如路径  $S \rightarrow B$  路径偏移度为  $\beta_2$ ,路径  $B \rightarrow A$  路径偏移度为  $\beta_1$ ,路径  $A \rightarrow A_0$  路径偏移度为  $\beta_0$ 。

可得出  $A_0$  伸展率为:

$$s = (r_0 + r_1 + r_2) / R = \sum_{i=0}^2 r_i / (\sum_{i=0}^2 r_i * \cos \beta_i) \quad (6)$$

同理可知,当网络规模为  $N$ ,最多分为  $M = \log_k N$  层时,任一节点  $X$ ,属于  $L_0, L_1, \dots, L_j$  层,其伸展率  $S_X$  为:

$$S_X = \frac{\sum_{i=j}^M r_i}{\sum_{i=j}^M r_i * \cos \beta_i} \quad (7)$$

为了计算网络链路上节点的平均压力,同样做近似分析。设网络的链路数为  $Q$ ,则  $L_i$  层中域的中心节点的链路数为  $Q / k^i$ 。这些链路上传输的同一数据包复制数  $\leq k * i$ ,则其平均压力为

$$\bar{\lambda} \leq (1/Q * \sum_{i=0}^{\log_k Q} Q * k * i / k^i) = k^2 / (k-1)^2 + O(\log Q / Q) \quad (8)$$

随着网络规模的扩大,节点增多,  $Q$  值趋于无穷时,平均压力以及最大压力分别可表示为

$$\bar{\lambda} = k^2 / (k-1)^2 \quad (9)$$

$$\lambda_{\max} = k \log_k N \quad (10)$$

从上述的分析中可以看出, TDRS 模型中域节点连通度很高,对于任一节点其控制平均开销为  $O(k)$ ,最坏情况为  $O(k \log_k N)$ ;而域内树的构建时间复杂度与 Prim 算法一致,为  $O(k^2)$ ,可见网络拓扑具有高聚集度、低路径长度的特性。

## 3 模拟仿真分析

本节对 TDRS 模型的数据传输性能进行仿真分析。考虑到大规模用户群的适用性,忽略了包丢失的情况。由 GT-ITM<sup>[7]</sup> 拓扑产生器初始生成 100 个节点,假设域间延迟为 30ms,域内延迟为 15ms,拓扑网络内成员的数目在 100 到 1000 之间变化,所有组成员在仿真时间 0 到 1000s 之间随机加入。另外,实验将这些节点按 NICE 协议构建了一棵多播树。每组仿真数据都是 3 次仿真结果的均值。

从图 3 可以看出,在拓扑网络节点数量从 100 增加到 900 的情况下, TDRS 模型最大链路压力的增加仅为 NICE 模型的 30%。从曲线的近似斜率可看出, TDRS 模型在节点数量大量增加的时候,对于链路压力膨胀有很好的抑制作用。

图 4 描述了多播组中平均时延的比较。从图中可以看出,节点数小于 300 时, TDRS 的平均链路压力大于 NICE。但随着多播组规模的扩展,节点数超过 600 时, TDRS 模型性能逐渐趋于稳定,平均链路压力显著降低。可见 TDRS 模型通过域饱和度产生目标域集,再通过综合参数平衡选取法确定目标加入域,它对链路压力的控制是有效的。

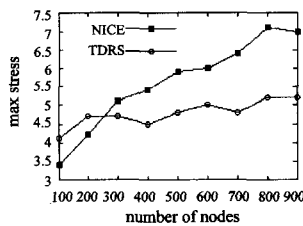


图 3 多播成员数 vs 最大链路压力

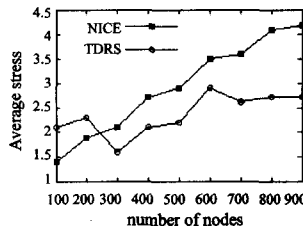


图 4 多播成员数 vs 平均链路压力

图 5 和图 6 描述了拓扑结构稳定后多播组传输时延的比较。 TDRS 模型虽然在拓扑网络构建初期会产生较大的控制开销,但在拓扑结构稳定后却能有效地降低时延。而 NICE 结构中,节点的入度数和出度数都比较高,所造成的传输延迟

也较大。当多播成员数目增加时,NICE 的数据传输效率显著降低。

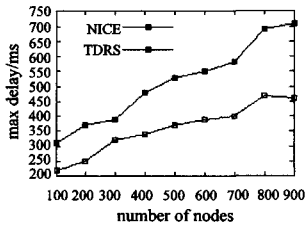


图5 多播成员数 vs 路径最大时延

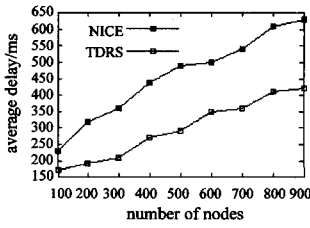


图6 多播成员数 vs 路径平均时延

以上分析说明,TDRS 模型同时而有效地平衡了传输时延和链路压力两个参数。NICE 模型由于 RP 负责对所有节点的控制,虽然在节点管理上有较好的特性,可在链路压力和时延的控制上却不够理想。TDRS 二维模型的提出,结合了 ALM 层次型与树型模型的优点,对于网络数据传输性能的改进意义深远。

**结束语** 本文提出了一种基于域饱和度的二维应用层多播模型,将同属于一个物理平面上的终端节点映射为网络二维逻辑结构。模型通过以域饱和度为基础的综合参数平衡选取法确定目标加入域,并运行带宽约束的最小半径支撑树算法 DCMR 来产生域内数据路由,同时采用基于优先级的动态规划法改变拓扑中各域中心节点所在的逻辑层次,以此达到 ALM 拓扑网络传输性能的整体优化。此外,模型利用节点冗余信息存储、链路数据预留来增强在网络拓扑异动下的自愈能力。

TDRS 致力于对拓扑网络数据传输通道性能进行优化,而对数据传输的安全关注不足。随着应用层多播技术应用的

日益广泛,加强多播通信安全是今后的一个重要研究方向。

## 参考文献

- [1] Francis P. Yoid; Extending the Multicast Internet Architecture. White paper. [http:// www. aciri. org/ yoid/](http://www.aciri.org/yoid/), 1999
- [2] Chu Y-H, Rao S G, Zhang H. A Case for End System Multicast // Proceedings of ACM SIGMETRICS. 2002, 20(8): 1456-1471
- [3] Banerjee S, Bhattacharjee B, Kommareddy C. Scalable Application Layer Multicast // Proceedings of ACM Sigcomm. Aug. 2002; 205-217
- [4] Prim R C. Shortest Connection Networks and Some Generalizations [J]. Bell System Technical Journal, 1957, 36: 1389-1401
- [5] Kruskal B. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem [J] // Proc. of the American Mathematics Society, 1956, 7(1): 48-50
- [6] Jin Xing, Wong Wan-Ching, Chan S-H G. High Performance Switching and Routing (HPSR) // 2005 Workshop. May 2005: 432-436
- [7] Zegura K C, Bhattacharjee S. How to Model an Internetwork // Proceedings of IEEE Infocom. 1996
- [8] Broash E, Shavitt Y. Approximation and Heuristic Algorithms for Minimum Delay Application-Layer Multicast Trees // Proc. of IEEE INFOCOM 2004. March 2004
- [9] Shi S Y, Turner J S. Multicast Routing and Bandwidth Dimensioning in Overlay Networks. IEEE Journal on Selected Areas in Communications, 2002, 20(8): 1444-455
- [10] Tan S W, Waters G. Building low delay application layer multicast trees // Merabti M, Pereira R, eds. Proc. of the 4th Annual PostGraduate Symp. The Convergence of Telecommunications, Networking & Broadcasting, EPSRC. 2003; 27-32

(上接第 53 页)

一个强度单位。从实验结果可以看出,最短路由算法 SPT 不管在什么情况下,链路的强度都为 1。而在其他两种算法中,链路的最大强度随着多播组中多播节点数量的增加而增大,但是 MaxUtility 算法增加的幅度要比算法 1 快得多,说明了本文实现的算法在实现网络效用最大化的同时,能够降低链路的强度,减少拥塞现象的发生,提高系统的可靠性。

**结束语** 本文主要研究了在无线 mesh 网络中联合优化效用和链路强度使网络的净效用最大化。在追求效用最大化的同时,降低链路的强度,避免链路带宽的浪费和减少拥塞现象的产生,提高系统的性能。同时,利用对偶分解法,获得一个新的基于价格的分布式算法,该算法联合优化了各个数据流的速率调整和流所经过的各条链路的强度。实验结果证实了算法的有效性。由于无线 mesh 网络是新兴起的一种宽带无线接入技术,还有许多问题需要解决,例如在无线 mesh 网络中有关覆盖多播的时延和多信道分配等问题有待于进一步研究,这是作者后继工作的研究重点。

## 参考文献

- [1] Akvildiz I, Wang F X. A survey on wireless mesh networks [J]. IEEE Radio Communications, 2005, 24(9): 23-30
- [2] Chu Y, Rao R, Zhang H. A case for end system multicast [J]. IEEE Journal on Selected Areas in Communications, 2002, 22(8): 1456-1471
- [3] De-Nian Y, Wanjiun L. On Bandwidth-efficient Overlay Multicast [J]. IEEE Transactions on Parallel and Distributed Systems, 2007, 18(11): 1503-1514
- [4] Cui Y, Xue Y, Nahratedt K. Optimal resource allocation in overlay multicast [J]. IEEE Transaction on Parallel and distributed Systems, 2006, 17(8): 808-823
- [5] Han D, Chan E. Quick Patch: an overlay multicast scheme for supporting video on demand in wireless networks [C] // 28th Int. Conf. Information Technology Interfaces (ITI). 2006; 613-618
- [6] Kim K. Bandwidth dependent overlay multicast scheme [C]. Communication Systems, ICCS, 2006; 1-5
- [7] Eren G, Kim A N. Utility optimal real-time multimedia communication in wireless mesh networks [C]. Packet video, 2007; 234-242
- [8] Bertsekas D. Nolinear Programming [M]. 2nd edition. Athena Scientific, 1999
- [9] Shor N Z. Minimization methods for non-differentiable functions [M]. Springer-Verlag, 1985