

一种主动网络环境下的主动队列管理机制

黎忠文¹, 李乐民², 李美蓉³

(1. 厦门大学计算机系, 福建厦门 361005; 2. 电子科技大学通信学院, 四川成都 610054;
3. 四川邮电学校, 四川成都 610064)

摘要: 针对主动网络这种新型网络体系能为主动队列的管理提供更加灵活和有效的方法, 以及目前主动网络中主动队列的管理仍然简单移用传统网络算法这一情况, 深入研究了这种简单移用存在的弊端, 及应专门为主动网络设计主动队列管理机制的重要性和必要性。在此基础上, 利用主动网络中路由器有着对流经其上的用户数据进行计算的特征, 提出了主动网络环境下, 基于资源动态协商的主动队列管理机制, 并用 OPNET8.0 进行了仿真实验, 验证了该机制在丢包率、队列长度和包延时等方面的优势。

关键词: 主动队列; 主动网络; 资源动态协商; 资源管理

中图分类号: TN915.01 文献标识码: A

Active queue management mechanism of active network

LI Zhongwen¹, LI Lemin², LI Meirong³

(1. Computer Department, Xiamen University, Xiamen 361005, China;

2. Communication College, university of Electronic Science and Technology of China, Chengdu 610054, China;

3. Sichuan Posts and telecommunications School, Chengdu 610064, China)

Abstract: Active network represents a new approach of network architecture which can provide a more flexible and effective management mechanism for active queue. In view of this and considering that traditional algorithms of active queue management are still used in active network, the shortcoming of using traditional algorithms in active network, importance and necessity of setting up a special active queue management algorithm for active network are studied. Active network's routers can perform computations on user data, while packets can carry programs to be executed in routers and possibly change their state. With this characteristic, an active queue management mechanism, based on dynamic resource trading, is put forward. A simulation that used to prove the advantages of the proposed mechanism in packet drop rate, length of queue and packet delay time is made under OPNET8.0 environment.

Key words: active queue; active network; dynamic resource trading; resource administration

1 引言

主动队列的概念最早由 IETF (Internet 工程任务组) 提出^[1], 并迅速得到了广泛应用。它作为路由器参与拥塞控制的手段, 其核心思想是在路由器中保持较小的队长, 即在该队列未满之前先丢弃报文, 发出拥塞指示, 从而加速拥塞的响应时间。常用的主动队列管理机制有随机提前检测 (random early detection RED) 及 WQ+ 2 等。这些队列管理算法在优化带宽、提高吞吐量和均衡带宽在不同流间的分配等方面都起到了较好效果。自 1995 年主动网络^[2] 概念问世以来, 人们在利用主动网络技术解决拥塞控制问题也都自觉和不自觉地使用了这些传统网络的主动队列管理算法, 如文献[2]研究的是主动网络环境下, 如何在单播通讯中实现拥塞控制, 提高 TCP

的性能; 文献[3]以支持分层多播作为例, 比较 ANTS 和 M0 两种主动网络结构的优劣; 而文献[4, 7]提出了一种主动网分层多播协议。在这些研究中, 解决拥塞控制的共同机制可归纳为一个消息单元和三个功能模块: 用于传送网络拥塞指示的消息; 网络主动节点用于网络拥塞的检测; 网络主动节点用于限制通过节点的数据传输; 网络数据源端用于限制数据的发送。它们的立足点是利用“主动网络”的可编程性, 即在主动节点上插入新的功能模块, 完成及时发送拥塞指示和拥塞解除消息的任务。它们对于主动队列这种与拥塞指示直接相关的数据结构的管理仍然采用传统网络的方法, 其中用得最多的是 RED 和加权 RED。由于传统网是基于“存贮-转发”特征而设计的, 如果在主动网络中, 对于向主动队列这种重要数据结构的管理, 不充分利用主动网络的优势而采用传统网络的

方法在某种程度上说是对主动网优势的“浪费”。最近有学者正着手为主动网络建立新的价格模型^[9]。本文首先讨论主动网络中采用传统网络算法存在的弊端;然后为主动网络设计了基于资源协商的主动网队列管理机制及实施的方案;最后通过仿真实验分析该机制的性能。

2 主动网络与传统网络

2.1 服务集

当用户包流经网络节点时,无论在主动网络还是传统网络环境下,网络都会提供一定的服务,以便把用户包从源端发送至目的端。令 N_T 和 N_A 分别代表传统和主动网络, $S_T = \{S_{T1}, S_{T2}, \dots\}$, $S_A = \{S_{A1}, S_{A2}, \dots\}$ 分别是传统和主动网网络每一跳所能提供服务的集合,其中 S_{Ti} , S_{Ai} , $i = 1, 2, \dots$ 是服务的类型。

基于“存贮-传发”的网络被称为传统网络,如 Internet 网。在分组交换网和面向连接的网中,交换设备主要的计算分别在于对分组头进行处理和执行信令协议,网络并不对用户数据内容进行调整和改动,因此每一跳的服务集可简化为 $S_T = \{S_{T1}, S_{T2}, S_{T3}\}$, 式中: S_{T1} —— 用于包交换或调度的服务; S_{T2} —— 用于包在网络节点上排队的服务; S_{T3} —— 用于包在链路上传输的服务。

主动网络则由一组被称为主动节点的网络节点构成,每个主动节点可以是路由器或交换机。主动网络特点:当用户包流经主动节点时,通过主动节点执行一定的可执行代码段,完成对用户包的处理,即网络中间节点可以对用户包的数据进行调整和改动。从主动网络的结构上看,可执行代码进入主动节点的方法分为三类^[13]:

(1) 主动包(active packet) 在主动包内封装有数据和可执行代码段,当主动包流经主动节点时,后者取出主动包内的可执行代码段并执行之,以实现主动包携带的数据进行处理,或者是改变主动节点的状态,以满足数据的需要。

(2) 执行代码事先注入主动节点,由主动包携相应的激活标志,以便当主动包流经主动节点时,主动节点能以主动包的激活标志为参数执行可执行代码段。

(3) 主动包携带轻量级的可执行代码,在主动节点上注入重量级的可执行代码,即当可执行代码段较短时由主动包携带,反之则事先注入主动节点,是第(1)、(2)种方法的折衷。当主动包流经主动节点时根据需要激活不同的可执行代码段。

因此与传统网络相比,主动网络中节点提供的服务丰富得多,所以其每一跳的服务集 $S_A = \{S_{A1}, S_{A2}, S_{A3}, S_{A4}\}$, 式中: S_{A1} 至 S_{A3} 分别与 S_{T1} 至 S_{T3} 类似; S_{A4} —— 主动节点执行可执行代码段对用户数据或节点本身进行相应处理的服务类。

2.2 传统网络算法应用于主动网络存在的弊端

令 C_{Np} , C_{Nq} , C_{Ni} 分别代表用户包在传统网络中某一跳获得的 S_{T1} , S_{T2} , S_{T3} 类服务所需花费,显然 C_{Np} , C_{Nq} , C_{Ni} 是相互独立的。在该跳,网络的总开销 $TC_T = \Phi(C_{Np}, C_{Nq}, C_{Ni})$; 然而如果 C_{Ap} , C_{Aq} , C_{Ai} , C_{As} 分别是用户包在主动网某一跳获得的 S_{A1} , S_{A2} , S_{A3} , S_{A4} 类服务所需花费,则 C_{Ap} , C_{Aq} , C_{Ai} , C_{As} 具有相关性,如对于一个视频点播而言,数据包在主动节点压缩

与否就让 C_{As} 与 C_{Ai} 相关。

算法都是基于一定服务集设计的,因此如果把传统网络算法应用于主动网络必然存在问题。

(1) 牺牲了主动网络作为“主动”的优势,网络利用率不高、服务质量差且缺少灵活性,不适于收费网络。(2) 对许多 Internet 新业务而言,如多媒体通信,流的特征是可变的。如一个流所需的带宽将在主动节点上,随着加压、解压和合并过程的执行而发生变化。这种可变流的存在,对网络的连接控制、资源的预留和路由等都有很大的影响。而传统网络关心的是对流的完整无缺的传送,它是基于流或包以固定的流量流经中间节点而设计的。

因此很有必要专门针对主动网络的特点设计相应的算法。

3 主动网络队列管理模型

3.1 主动网络基于资源动态协商的管理机制

自开放系统问世以来,基于资源协商(又称为基于市场)^[8]的管理机制一直是研究的热点,其优势在于它有望解决“非集中式管理”、“引导用户合理使用资源”和“QoS 问题”等当前大型系统所面临的难题。该机制的核心思想是为网络资源定价格(一般价格不可变),然后系统按一定的规则把资源分配给用户。

主动网络中,为了更好地反映网络资源使用情况的实时性和分配的公平性,采用资源价格可变的管理措施,既资源的价格是当前可用资源的函数 ζ , 该函数值随网络当前可用资源的减少而增加。

设 X_1, X_2, \dots, X_n 分别为用户包享受主动网某一跳的服务时,所消费资源 R_1, R_2, \dots, R_n 的数量; P_1, P_2, \dots, P_n 分别是这些资源的“市场”单价; I 为用户在该跳总消费预算。则资源分配方案可以是:

(1) 在 I 限制下享受尽可能多的服务。如在分层多播应用中,传送尽可能多的高层数据,以获得优质声音和图像。于是用户包获得的网络资源 X_1, X_2, \dots, X_n 应满足下列式子

$$\text{MAX} (X_1 \times P_1 + X_2 \times P_2 + \dots + X_n \times P_n \leq I)$$

约束条件

$$X_i \leq X_{i\max}, i = 1, \dots, n$$

式中: $X_{i\max}$ —— 每个用户包在主动网每一跳可用资源 R_i 的最大限制。大多数主动网络都有这个限制,如而 SNAP^[11] 则明确给出该限制。

(2) 在 I 和 $X_{i\max}(i = 1, \dots, n)$ 限制下尽可能节省开支

$$\text{MIN} (X_1 \times P_1 + X_2 \times P_2 + \dots + X_n \times P_n)$$

约束条件

$$X_1 \times P_1 + X_2 \times P_2 + \dots + X_n \times P_n \leq I$$

$$X_i \leq X_{i\max}, i = 1, \dots, n$$

(3) 在 I 和 $X_{i\max}(i = 1, \dots, n)$ 限制下,设置各资源之间的比重,即如果用户更加偏重于使用哪种资源,则设置在该资源上的权就大,因此用户包获得该资源的概率就越大。

3.2 基于资源动态协商的主动网络队列管理模型

网络路由器中等待链路用的队列有多个,如多优先级队

列管理等。为了简化说明,不妨把它们合为一个,其输入/输出端分别连接到路由器的输入/出链,于是传统网络等待链路的队列模型可以简化为图 1 所示。



图 1 传统网络队列模型

虽然资源系统有多种,如文件系统、数据库、CPU 占用时间、存储器与带宽等都是系统的资源,但主要的底层资源是 CPU 占用时间、存储器与带宽 3 种^[5]。和文献[6]一样,这里把 CPU 和存储器的花费放在一起考虑。为了在主动网络中用户可以在系统资源中进行有选择的使用或让系统管理员合理的分配系统资源,我们在主动节点中另开辟一组缓冲区,它可是队列也可以是堆栈。本文主要讨论思想,因此不妨把它也视为一个队列,于是主动网络的队列模型如图 2 所示。其中队列 1 为传统网络中的队列(等待链路用),队列 2(包等待 CPU 和存储器,以接受 SA4 类服务)为主动包进行资源协商所采用。当主动包进入路由器后,可以有多种算法来选择进入队列 1 还是队列 2,典型的算法有

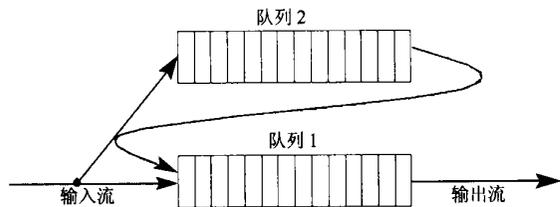


图 2 主动网络的队列模型

(1) 主动包分别以 λ_x 、 λ_y 的概率进入两个队列。这种算法适用于用户采用的资源使用算法为前述的第三种,即按一定的比例消费资源。

(2) 分别计算队列 1 和队列 2 的当前价格,然后计算本主动包在这一跳的花费,最后进入最便宜的队列。注意这时进入队列 2 的花费为主动包消费队列 2 的花费加上从队列 2 出来进入队列 1 后所需的花费之和。

(3) 主动包总是优先考虑进入队列 1,只有当队列 1 的价格超过一定预算时,使用上述第 2 种算法。

队列价格函数在仿真实验中,和文献[6]一样,设计为凸增函数,以保证队列的价格随队列的负载的增加而激速上升。

3.3 仿真实验

3.3.1 仿真环境

仿真实验是在 OPNET8.0 环境下进行的。为了更好的说明问题,我们采用尽可能简单的实验。仿真采用图 3 所示的网络拓扑。主动节点中设有二个队列,其中队列 1 用于等待输出链路,队列 2 用于资源协商,其上的操作设为对包进行压缩的算法;主动包通过主动节点从源 $S_i(i=1, \dots, n)$ 发往目的端 $R_i(i=1, \dots, n)$ 。实验中设置了两个源 S_1 和 S_2 , 它们的包长都服从均长度 1 000 bps 的指数分布, S_1 包间间隔为 0.5 s, S_2 包间间隔足均长为 1 s 的指数分布;链路均为 4 000 bps;主动节点中队列 1 的容量是 60 000 b, 队列 2 的容量是 30 000 b;队 1、队 2 服务率均为 1 800 1/s;队列

的当前价格是其负载的函数,队列 1 的价格 $price_1$ 和队列 2 的价格 $price_2$ 计算公式分别设为

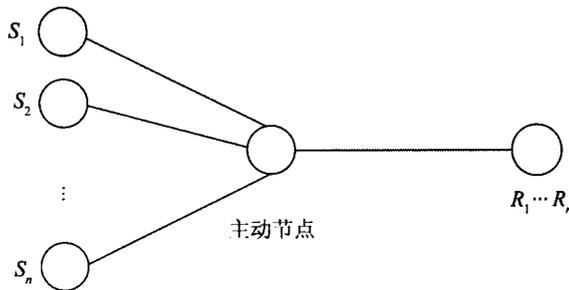


图 3 仿真拓扑结构图

$$price_1 = 1000 \times \sqrt{\frac{1.01}{1.01 - load_1}} - 999$$

$$price_2 = 1000 \times \sqrt{\frac{1.01}{1.01 - load_2}} - 999 + 180$$

式中: $Load_1$ 和 $Load_2$ ——两队列当前的长度与队列容量之比;设置 180 是为了让主动包能优先利用链路资源,它也可设为其它值。

实验分为两个方案:

- 方案 1 传统的主动队列管理,队列 2 失效;
- 方案 2 主动网络的队列管理,队列 2 使能。

主动队列的管理算法为 Tail Drop 法。仿真时间为 100 s。主动包入队的算法选用 3.2 节中的第 2 种和第 3 种算法的折衷,并设 SA4 设为压缩服务, β 是压缩比(实验中 $\beta = 0.6$):

```
while (包 p 到达) do
{
    if (p 来自队列 2)
    then
    {
        p 进入队列 1;
        Load2= (len(队列 2)- len(p))/30 000;
    }
    else
    {
        计算 price1 和 price2;
        cost1(p)= price1 * len(p);
        cost2(p)= price2 * len(p) + price1 * len(p) * beta;
        if (min(cost1(p), cost2(p)) > I)
        { 结束本次 while 循环; }
    }
    if (cost1(p) < cost2(p))
    then
    {
        p 入队列 1;
        Load1= (len(队列 1)+ len(p))/60 000;
    }
    else
    {
```

p 入队列2;

$$Load_2 = (\text{len}(\text{队列}2) + \text{len}(p)) / 30\ 000;$$

$$Load_1 = (\text{len}(\text{队列}1) + \text{len}(p) \times \beta) / 60\ 000;$$

其中: $\text{len}(\text{队列}1)$ 、 $\text{len}(\text{队列}2)$ 和 $\text{len}(p)$ ——队列1、队列2和包 p 的长度,单位为 bit; $\text{cost}_1(p)$ 和 $\text{cost}_2(p)$ 分别指包 p 进入队列1和2的花费; I 是包 p 在本跳的消费预算。

3.3.2 仿真结果与分析

从图4可见方案1中,经过约30s以后队列1的长度不断地增加,以至队满,可以得出存在大量丢包的结论。而在方案2中,由于队列1长度的增加,使其价格也上涨,有些主动包就直接进入队列2,因此方案2中队列1的瞬时长度比方案1短,这与算法相符,且在该方案中队列1和2均未满,因此也可得出方案2中不存在丢包的结论。

通过在目的端对所收包进行分类统计,得到了图5和图6。图5表明方案1存在大量的丢包,而方案2则没有丢包发生。图6表示目的端收到包的数量,无论对于源 S_1 还是源 S_2 而言,在约40秒以后方案1比方案2收到的同源包逐渐明显减少,这与图4和图5相符。图5和图6充分说明了图4所得结论的正确性。

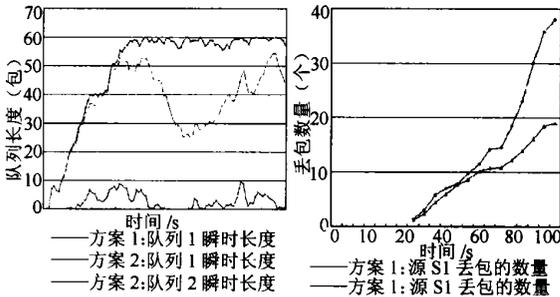


图4 方案1和方案2队列瞬时长度比较图

图5 丢包数量与时间的关系图

图7是包延迟与时间关系图。Y轴是X轴相应时刻产生的包的延迟时间。事实上该图应为点线图,只是为了更好的看出其趋势,才把它们连为线。从图7可见方案2的包延迟时间平均小于方案1,这是因为方案2中有大量包进入了队列2,得到了压缩处理,是合理的。这说明在工程中,可以通过在主动节点上合理分配CPU来设置队列的服务率以达到减少包延迟时间的目的。

4 结论

主动网络 and 传统网络是两类不同的网络,前者基于网络可编程性,允许可变流;后者则是“存贮-转发”类网络,基于固定流量。针对目前传统网络算法被简单移用于主动网络的情况,本文研究了该简单移用存在的弊端,并以主动队列为对象设计了基于资源动态协商的主动网络的管理算法,然后用 OPNET8.0进行了仿真实验,说明了专门为主动网络设计算法的优势。我

们下一步的研究计划是进一步完善本算法,并最终加入到主动网络SNAP中,以增强SNAP的资源管理策略。

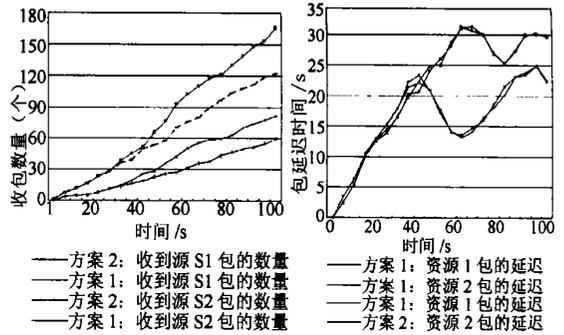


图6 收包数据与时间关系图

图7 包延迟与时间的关系图

参考文献:

- [1] Braden B. Recommendation on Queue Management and Congestion Avoidance in the Internet[J]. RFC309, 1998.
- [2] Faber T. ACC: Using Active Networking to Enhance Feedback Congestion Control Mechanisms[J]. IEEE Network (Special Issue on Active and Programmable Networks), 1998:61-65.
- [3] Banchs A. Multicasting Multimedia Streams with Active Networks[C]. Proc. of IEEE Local Computer Networks Conference, LCN'98, Boston, USA, 1998. 150-159.
- [4] Yamamoto Lidia, Leduc G. An Active Layered Multicast Adaptation Protocol[C]. Second International Working Conference on Active Networks, 2000. 180-194.
- [5] Tschudin C. Open Resource Allocation for Mobile Code[C]. Proceedings of the Mobile Agent'97 Workshop, Berlin, Germany, 1997.
- [6] Yamamoto Lidia, Leduc G. An Agent-Inspired Active Network Resource Trading Model Applied to Congestion Control. Second International Workshop on Mobile Agents for Telecommunication Applications, 2000. 151-169.
- [7] Keller R. An Active Router Architecture for Multicast Video Distribution[C]. Proc. IONFOCOM 2000, 2000.
- [8] Ferguson D F. Economic Models for Allocating Resources In Computer Systems, in Market Based Control: A Paradigm for Distributed Resource Allocation[M]. Scott Clearwater, World Scientific Press, 1996.
- [9] Najafi K, Garcia L V. A Novel Cost Model for Active Networks. Proc. of Int. Conf. on Communication Technologies, World Computer Congress, 2000.
- [10] Wetherall D, Guttag J, Tennenhouse D L. ANTS: A Tool Kit for Building and Dynamically Deploying Network Protocols[C]. IEEE OPE-NARCH'98, San Francisco, 1998.
- [11] Moore J T. Safe and Efficient Active Packets. Technical Report MS-CIS-99-24[R]. Department of Computer and Information Science, university of Pennsylvania, 1999.
- [12] DARPA. DARPA Active Network Home Page[EB/OL]. http://www.darpa.mil/to/research/anets/index.html, 2002.
- [13] Tennenhouse D L. A Survey of Active Network Research[J]. IEEE Communications Magazine, 1999: 80-86.