基于多种资源测量的流媒体实时接纳控制研究

杨 帆，张冬松，金士尧，卢 为
（国防科学技术大学 计算机学院 并行与分布处理国防科技重点实验室，长沙 410073）

摘 要：为了根据计算、存储资源和网络带宽的占用情况进行更加精确的接纳控制决策，设计了受限的可变时间窗口测量法，提出了基于多种资源占用情况的测量和算法（MBMS）。在获取多种资源负载情况的基础上实现了流媒体服务的实时接纳控制方法。实验结果表明，该方法能够在高负载的情况下确保较高的服务接纳率，同时在低负载情况下提高了资源利用率。

关键词：多种资源；动态测量；流媒体服务；接纳控制
中图分类号：TP309 **文献标志码**：A **文章编号**：1001-3695(2012)03-1044-04
doi:10.3969/j.issn.1001-3695.2012.03.066

Research on multi-resource-measurement-based admission control for real-time streaming media services

YANG Fan, ZHANG Dong-song, JIN Shi-yao, LU Wei
(*Key Laboratory of Science & Technology for National Defence of Parallel & Distributed Processing, School of Computer Science, National University of Defense Technology, Changsha* 410073, *China*)

Abstract: To make highly accurate decisions for the admission control based on the consumption of computing, storage resources and network bandwidth, this paper proposed the dynamic measurement using constrained time window, and presented the multi-resource-based measured sum algorithm (MBMS). It implemented the admission control for real-time streaming media services depending on the obtainment of the system load. Experimental results show that this method ensures a high acceptance rate of the service in the high-load conditions, while improves resource utilization in the low-load case.

Key words: multi-resource; dynamic measurement; streaming media services; admission control

流媒体应用具备典型的软实时特点，近年来发展迅速。随着服务质量的不断提高，流媒体应用占用的各类资源数量越来越大。通过测量获取计算、存储资源和网络带宽的变化情况，利用接纳控制机制调节系统负载，解决有限资源与实时任务时限要求之间的矛盾，具有现实意义[1]。

接纳控制是网络服务质量（quality of service，QoS）的重要组成部分[2]，分为基于参数的接纳控制（parameter-based admission control，PBAC）和基于测量的接纳控制（measurement-based admission control，MBAC）两种。PBAC 要求给出流的特征并建立数学模型[3]。由于互联网中流特征复杂，加之受长程相关和自相似性影响[4,5]，难以精确建模。MBAC 无须预知流量模型，只需测量网络流量即可作出接纳判断，在实际应用中能够达到更高的资源利用率[6]。此外，还有基于策略的接纳控制[7]和基于拥塞预处理的接纳控制[8]方法。前者着力于决策阶段用的调度机制，未考虑测量阶段的改进；后者可以纳入 MBAC 范围内，在测量阶段用预先发送探测包以获取链路状态，但是探测包与数据包所经路由不尽相同，探测过程容易增大网络负载，导致该方法实际效果不佳。另外，文献[9]提出了基于自相似测量的自适应接纳控制算法 HSS-AMBAC。当网络流量较大时，若网络流量在设定阈值附近波动，HSS-AMBAC 算法会在短时间内变得极为保守，导致接纳能力大幅降低并且难以恢复；当网络流量较小时，该算法的测量开销显著增大。

与现有研究大多局限于网络带宽不同，本文提出了一种受限的可变时间窗口测量法，实现了对计算、存储资源和网络带宽的动态测量；基于此，设计了一种基于多种资源占用情况的测量和算法（multi-resource-based measured sum，MBMS），并实现了流媒体的实时接纳控制方法。

1 受限的可变时间窗口测量法

流媒体应用是指音视频在网络上以实时的、无须下载等待的方式进行播放[10]，如实时视频会议、在线新闻直播、远程医疗、视频点播等。流媒体应用具有软实时性，基本特征是：用户提出请求之后，系统应当尽快响应，确保以既定的服务质量在指定时限内完成[11]。与硬实时系统相比，流媒体应用能够容忍更大程度的性能下降，适应性更强，网络利用率更高。

1.1 测量对象

以前的网络测量研究仅关注网络带宽，忽略了其他资源对接纳控制的影响。然而，现有的视频文件采用多种编码格式和封装方式，处理过程中消耗的计算资源越来越多；视频时长动辄数十分钟，占用的存储空间越来越大；高分辨率视频的出现，直接导致网络带宽的占用率越来越高。随着流媒体技术的成

收稿日期：2011-06-22；**修回日期**：2011-07-25

作者简介：杨帆（1984-），男，内蒙古巴彦淖尔人，硕士，主要研究方向为计算机网络与通信、实时系统（yangfan1944@163.com）；张冬松（1980-），男，河南信阳人，博士研究生，主要研究方向为实时系统；金士尧（1937-），男，江苏苏州人，教授，博导，主要研究方向为实时系统、分布式交互仿真、虚拟现实等；卢为（1984-），男，湖南长沙人，硕士研究生，主要研究方向为实时系统.

熟和发展,用户对服务质量提出了更高要求,在接纳控制中仅考虑单一因素显然不够,而应该综合考虑计算、存储资源以及网络带宽的占用情况,才能作出准确判断。

1.2　测量机制

本文兼顾计算、存储和网络带宽等资源,设计了一种受限的可变时间窗口测量法。其中,测量窗口 T 和采样时间 S 最为关键。T 的值决定了算法受历史数据影响的程度。T 值越小,得出的估计负载越符合实际情况,系统的接纳能力越强,资源利用率越高,但是测量开销会增大;T 值越大,算法越保守,系统的接纳能力越小。采样时间 S 决定了算法对负载变化的敏感程度,数值过大会降低算法精度,数值过小同样会增大测量开销。

令 $T=nS(n\in N,N$ 为正整数集)。设 L_T 为测量窗口内的估计负载,L_S 为采样时间内的平均负载,增设触发 T 发生改变的资源占用利用率上限为 L_{upper}、下限为 L_{lower},以及 T 的变化区间$[T_{min},T_{max}]$ 和默认值 T_0,以实现对计算资源、存储资源和网络带宽的动态测量。

规则 1　当 $L_T>L_{upper}$时,逐渐增大 T,使算法变得更保守,减小接纳能力,但要确保 $T\leqslant T_{max}$。

规则 2　当 $L_T<L_{lower}$时,逐渐减小 T,使算法更精确地反映负载的实际变化,但要保证 $T\geqslant T_{min}$。

规则 3　当 $L_{lower}<L_T<L_{upper}$时,令 $T=T_0$,防止因 T 长期偏离默认值导致系统接纳能力下降。

估计负载 L_T 则通过下面的规则得到。

规则 4　测量窗口 T_i 内的 L_T 的初值为上一个测量窗口 T_{i-1}中 n 个平均负载的最大值。即 $L_T=\mathrm{MAX}(L_{S_1},L_{S_2},\cdots,L_{S_n})$,其中,$S_1,S_2,\cdots,S_n$ 为 T_{i-1}内的 n 个采样时间。

规则 5　若采样时间 S_j 内的平均负载 $L_{S_j}>L_T$,则 $L_T=L_{S_j}$。

规则 6　若新服务接入系统,则置 $L_T=L_T+R$,R 表示新到服务所需的资源占用量。

规则 7　若服务完成后离开系统,则置 $L_T=L_T-P$,P 表示完成的任务的资源占用量。

1.3　测量方法

网络资源测量过程中,采样频率高,循环次数多,累计的测量开销不容小视。此外,系统负载接近饱和时,额外的测量开销容易引起资源竞争,甚至导致拥塞或服务终止。如果使用用户态的应用程序进行测量,在用户态与核心态切换以及文件操作时都会产生较大开销。为了降低测量开销,必须寻找更好的实现方法。可加载内核模块(loadable kernel module,LKM)能够动态地向 Linux 内核添加或删除代码,实现了最小的内存占用。LKM 运行在内核模式,比用户态的应用程序具备更高的访问权限,运行期间造成的资源开销也比较小。

Proc 文件系统是 Linux 中的一个特殊的虚拟文件系统,为访问系统内核数据的操作提供接口,向用户和应用程序提供系统中进程运行的各种信息。Proc 中的文件只存在于内存当中,不占用外存空间,在用户或应用程序读取 Proc 文件时动态地从内核中生成,具有很强的实时性。

在 Linux 包含的多种虚拟文件系统中,Proc 的历史最久,应用最广。Linux 内置的监控工具(如 top、free、ps、netstat 等)和第三方软件商开发的系统监测软件(如 Gnome System Monitor、KDE System Guard 等),都利用了 Proc 的动态性好、开销较小等特点。本文结合 Proc 和 LKM 的优点,实现了对计算资源(CPU)、存储资源(内存)、网络带宽资源等的动态测量。

网络资源可用余量的测量过程如图 1 所示。通常采样时间 S 设为几百个分组的发送时间,本文将 S 的初值设为 1 s。测量窗口的初值 T_0 为 10 s。测量过程分为数据获取和负载估计两个阶段,在时隙控制模块的指挥下完成。具体描述如下:

a)数据获取阶段开始后,首先加载内核模块读取 Proc 系统内的相关文件,计算采样时间 S 内的计算、存储资源和网络带宽的占用情况:

(a)计算 CPU 占用率 $\hat{c}$。CPU 总时间从 stat 文件获取,包含 7 个参数,单位为 jiffies,公式如下:

$$C_{total}=\mathrm{user}+\mathrm{system}+\mathrm{nice}+\mathrm{idle}+\mathrm{iowait}+\mathrm{irq}+\mathrm{softirq}$$

其中:user 表示用户态的 CPU 时间(不包含 nice 值为负的进程);system 表示核心态时间;nice 表示 nice 值为负的进程所占用的 CPU 时间;idle 表示除硬盘 I/O 等待时间以外的其他等待时间;iowait 表示硬盘 I/O 等待时间;irq 表示硬中断时间;softirq 表示软中断时间。在两个间隔较短的时刻 idle_1、idle_2 进行测量,求得 CPU 的瞬间利用率。因此,$\hat{c}=(1-\mathrm{idle}/C_{total})\times100\%$。

(b)计算内存占用量 $\hat{m}$。Meminfo 文件中包含内存总量 M_{total}和内存空闲量 M_{free}。因此,$\hat{m}=M_{free}$。

(c)计算网络带宽占用量 $\hat{v}$。Net/dev 文件包含网卡数据接收量 N_{rece}和发送量 N_{tran}(单位:Byte)。目前网络设备接口大多支持全双工模式,计算数据传输率时应当对上行(接收)和下行(发送)传输率进行区分。在流媒体应用中,服务器端的下行数据高于上行数据两个以上数量级。因此在决策过程中主要关注下行速率对服务质量的影响。

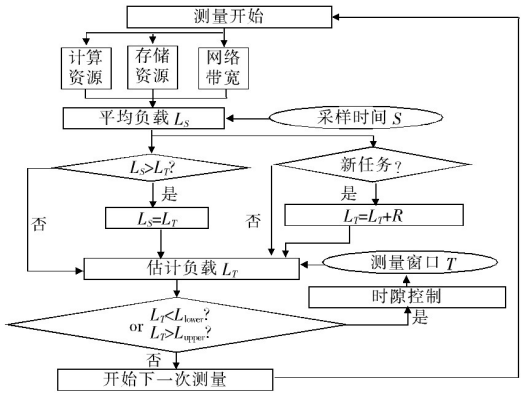


图 1　网络资源可用余量的测量过程

按时间间隔 t 先后两次测量 N_{tran}的值 N_{t1} 和 N_{t2},则占用的网络带宽为

$$\hat{v}=8(N_{t2}-N_{t1})/t\quad(\text{单位:bps})$$

b)获取计算、存储资源和网络带宽的占用量后,以 T 为周期,按规则 4、5 计算系统的估计负载。本文将三种资源的占用上限 L_{upper}与资源总量的比值限定在 0.8,将占用下限 L_{lower}与资源总量的比值限定为 0.2。若估计负载超出(L_{lower},L_{upper})区间,时隙控制模块会减小或增大 T 的值,使算法变得激进或保守。规则 6、7 在流媒体服务器内实现,详细说明见 2.2 节。

2　基于多种资源测量的流媒体传输接纳控制方法

传统的 MBAC 分为测量和决策两个阶段。决策阶段采用的方法很多,如 Hoeffding bounds(HB)、Tangent at peak(TP)、measure CAC(MC)、aggregate traffic envelopes (TE)等。Breslau

等人[12]对比了五种 MBAC 算法,发现尽管各算法所用参数和公式均不相同,但性能没有明显差异。考虑到在流媒体应用中,单个任务对计算、存储资源和网络带宽的占用量容易估计,并且服务等级相同的任务所占用的资源数量大致相同,因此,本文提出一种基于多种资源占用情况的测量和算法(MBMS)来实现接纳控制决策。该算法原理简单,可以进一步降低计算开销。

2.1　基于多种资源占用情况的测量和算法(MBMS)

流媒体应用具有典型的软实时性特点,要在确定时限内完成只允许可接受范围内的服务质量下降。随着用户对服务质量要求的提高,流媒体应用所需的资源量迅速增长,而网络中共享的计算、存储资源和网络带宽是有限的。如果在系统负载接近饱和时依然接受新任务,会造成画面失真、丢帧,播放停顿甚至任务中止。如果系统负载较低,可以适当调整服务质量,向用户提供更好的服务,提高资源利用率。MBMS 算法能够根据资源占用情况,作出接纳控制判断,并能实时调整系统负载。

MBMS 算法的决策依据如下所示:

$$\begin{cases} \hat{c} + c^r < \varphi_c C & (1) \\ \hat{m} + m^r < \varphi_m M & (2) \\ \hat{v} + v^r < \varphi_v V & (3) \end{cases}$$

在共享网络中,资源利用率接近饱和会引发竞争,导致服务质量下降,因此要限制每种资源的最大利用率 φ,本文限定 $\varphi = 90\%$,不同种类的资源用下标进行区分。

式(1)表示依据计算资源进行的判断。其中:$\hat{c}$ 为测量得出的 CPU 利用率;c^r 为新到服务 R 请求的 CPU 占用率;C 为系统的 CPU 总利用率,即 $C = 100\%$;φ_c 为系统定义的最大 CPU 利用率。

式(2)表示依据存储资源进行的判断。其中:$\hat{m}$ 为测量得出的存储器占用情况;m^r 为新到服务 R 请求的存储器占用量;φ_m 为系统定义的最大存储器利用率;M 为系统的存储器总量。

式(3)表示依据网络带宽进行的判断。其中:$\hat{v}$ 为测量得出的带宽占用量;v^r 为新到服务 R 请求的带宽;V 为系统的网络带宽总量;φ_v 为系统定义的最大带宽利用率。

若新到请求所需的资源量与系统负载之和同时满足上述三个不等式,接纳该服务请求;否则拒绝。

2.2　控制方法

目前,流媒体应用趋向于根据用户需求进行差别化服务。本文将服务质量划分为高、中、低三个服务等级,分别对应于高清、标清、普通三种传输标准。表 1 为不同分辨率的流媒体服务在服务器端所占用的资源数量。

表 1　不同等级流媒体传输的资源占用情况

分辨率/ppi	CPU 占用率/%	存储器占用量/MB	带宽占用量
高(≥1200×720)	2.0 ~ 3.5	12 ~ 20	1 ~ 2 Mbps
中(≈800×600)	1.0 ~ 2.0	5 ~ 10	256 ~ 800 kbps
低(≤480×320)	0.5 ~ 0.8	≤5	≤100 kbps

接纳控制的决策流程如图 2 所示,各个步骤的具体操作如下:

a)新任务到达系统时,按中等服务水平进行第一次接纳控制判断。若此时测量给出的估计负载超过设定上限,则拒绝当前请求;若当前估计负载小于规定上限,则进入步骤 b)。

b)按 1.2 节中规则 6 进行计算,若当前的估计负载与新任务请求的负载量之和超过上限,拒绝该任务;否则接受任务。用计算结果更新测量模块中的估计负载并进入步骤 c)。

c)查看新任务队列,如果还有更多新任务,进入步骤 a);否则,根据系统负载情况调整系统内任务的服务等级:若当前负载偏小,即 $L_T < L_{lower}$,进入步骤 d);若当前负载偏大,即资源利用率 > φ,进入步骤 e)。

d)按任务持续时间递增的顺序依次提高系统内任务的服务等级。将原来的低等服务等级调整为中等,中等服务等级调整为高等。每次完成调整后,用计算结果更新测量模块中的估计负载,直至 $L_T > L_{lower}$;若有任务结束,进入步骤 f)。

e)按任务持续时间递增的顺序依次降低系统内任务的服务等级。将原来的高等服务等级调整为中等,中等服务等级调整为低等。每次完成调整后,用计算结果更新测量模块中的估计负载,直至资源利用率 < φ;若有任务结束,进入步骤 f)。

f)任务结束时,按规则 7 更新测量模块中的估计负载。

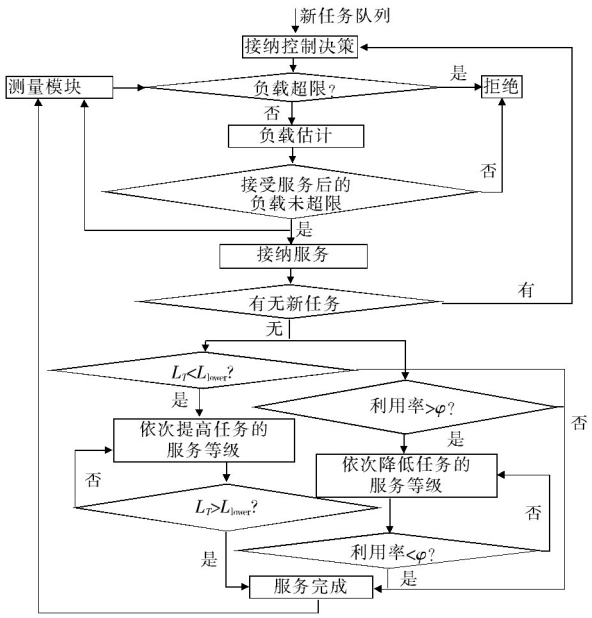


图 2　接纳控制流程

3　实验

3.1　实验环境

实验平台采用 C/S 结构,由 1 台服务器和 5 台客户机组成。服务器所用 CPU 为英特尔酷睿 2 Q8200 四核处理器,主频 2.33 GHz;内存 1 GB;网卡工作模式为全双工,传输速率 100 Mbps;操作系统为 Redhat Enterprise Linux 4,内核版本 2.6.9;安装 Apple 公司的 Darwin Streaming Server(DSS)作为流媒体服务器软件。DSS 是 QuickTime 流服务器的开源版本,为用户和开发人员预留 QTSS 编程接口,最新的版本是 6.0.3。

5 台客户机分别采用英特尔和 AMD 的双核处理器,主频均大于 2.0 GHz;内存 1 GB;网卡工作模式为全双工,传输速率 100 Mbps;操作系统为 Windows XP 和 Windows 7 basic,并安装 Apple 公司开发的 QuickTime 流媒体播放软件。

服务器和客户机用路由器连接,工作模式为全双工,接口速率 100 Mbps。服务器所用流媒体文件共分为高清、标清和普通三类,按视频时长分为长片和短片两种。表 2、3 列出了实验所用视频的具体参数。

表 2　实验所用短片参数

短片类别	码率/kbps	大小/MB	时长/min	数量/个
高清	717 ~ 840	187 ~ 350	10 ~ 20	20
标清	501 ~ 532	68 ~ 134	3 ~ 15	20
普通	212 ~ 253	27 ~ 35	3 ~ 10	20

表 3　实验所用长片参数

长片类别	码率/kbps	大小/MB	时长/min	数量/个
高清	717 ~ 840	902 ~ 1244	30 ~ 50	5
标清	501 ~ 532	600 ~ 1000	80 ~ 120	15
普通	212 ~ 253	300 ~ 500	80 ~ 120	10

3.2　实验方法

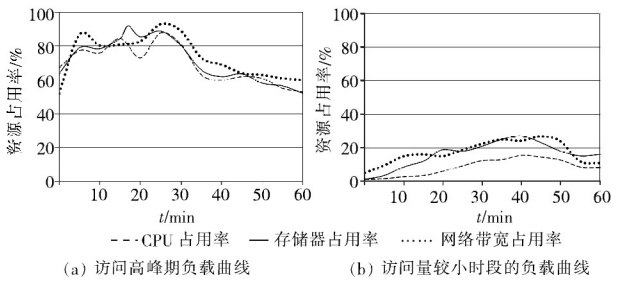
在 Linux 系统中实现了第 1 章描述的动态测量内核模块。在 Darwin Streaming Server 中增加了第 2 章描述的接纳控制模块。

采样时间为 1 s,测量窗口默认为 10 倍的采样时间。动态测量模块以默认的测量窗口长度(10 s)为周期监测计算、存储资源和网络带宽的占用情况,当三者中任何一个的占用率超过了 80%,则使下一个测量窗口增加一个采样时间;当三者的占用率全部低于 20%,则使下一个测量窗口减小一个采样时间。

接纳控制模块从动态测量模块获取三种资源的占用数据。新任务到达时,按中等服务等级所需资源量进行接纳控制决策,若任一资源的占用率超过 90%,或者新任务所需资源与当前负载之和超过 90%,拒绝该请求。当资源占用量超过系统的最大限制时,降低系统内若干任务的服务等级,减小负载;资源占用率均小于 20% 时,提高系统内若干任务的服务等级,提高资源利用率。

3.3　结果分析

客户机利用 QuickTime 随机访问服务器上的多媒体文件。实验时间持续 48 h,根据测量结果绘制出共享资源利用率曲线图。本文发现计算、存储资源和网络带宽的利用率以 24 小时为周期呈现循环的趋势。从访问高峰时段和访问量较小时段中各截取 1 h 的数据,绘制资源利用率随时间的变化曲线,如图 3 所示。MBMS 算法在高负载情况下的服务接纳率达到了 51.21%。从图 3(a)可以看出,通过限制接入系统的服务数量,MBMS 将计算、存储资源和网络带宽在高负载下的占用率限制在实验设定的上限(90%)附近,避免了因资源耗尽而导致任务超时或中止。在图 3(b)所示的低负载情况下,服务接纳率接近 100%。



(a) 访问高峰期负载曲线　　(b) 访问量较小时段的负载曲线

图 3　资源利用率曲线

图 4 为计算、存储资源和网络带宽占用率与测量窗口之间的关系。当计算、存储资源和网络带宽任一资源的占用率超过 80% 时,MBMS 算法会逐渐增大测量窗口以减小系统接纳能力。从图 4 中可以看出,三类资源的占用率大于 80% 后,测量窗口增长幅度较小时,三类资源的占用率仍有上升趋势;当测量窗口增大至 13 S ~ 15 S 后,三类资源的占用率才逐渐下降。其原因是测量窗口增长较小时,对系统的接纳控制能力影响不明显;当窗口增长幅度较大时,接纳能力明显下降,三类资源的占用率也随之下降。

如图 4 所示,测量窗口从 10 S 减小到 5 S 的过程中,三类资源的占用率在总体上呈现降低的趋势,但是会有部分曲线出现波动甚至上升。其原因是三类资源的占用率较低时,系统内运行的任务数量较少,MBMS 算法通过逐步减小测量窗口可以提高接纳能力,如果有新服务被接受,则会提高资源占用率。

4　结束语

本文提出的 MBMS 算法采用了受限的可变时间窗口测量法,能够动态获取 CPU、内存、网络带宽的占用情况,并能以之为依据进行有效的接纳控制决策。算法中的 T、S、L_{lower}、L_{upper}、φ、T_{min}、T_{max}、T_0 等参数的取值允许调整,可以适应不同平台、不同水平的性能需求。实验结果表明,MBMS 算法能够在高负载情况下实现较高的服务接纳率,在低负载情况下提高资源利用率,是流媒体应用中的实时接纳控制技术研究的一次有益探索。

参考文献:

[1] 金士尧. 共享资源网络的实时任务接纳控制方法:中国,ZL200810144000.3[P]. 2010-12-01.

[2] NAM S Y, KIM S, SUNG D K. Measurement-based admission control at edge routers[J]. IEEE/ACM Trans on Networking, 2008, 16(2): 410-423.

[3] 翁苏湘, 金永生, 王彦丽. 网络资源分配中的接纳控制技术[J]. 吉林大学学报, 2008, 26(6): 609-614.

[4] BERRAN J. Long range dependence[J]. Wiley Interdisciplinary Reviews: Computational Statistics, 2010, 2(1): 26-35.

[5] KARAGIANNIS T, MOLLE M, FALOUTSOS M. Long-range dependence: ten years of Internet traffic modeling [J]. IEEE Internet Computing, 2004, 8(5): 57-64.

[6] 刘俊, 陈昊鹏. 基于 QoS 延时参数的服务接纳控制算法[J]. 计算机工程, 2008, 34(19): 89-91.

[7] GONCALVES P, AZEVEDO R, GOMES D, *et al*. Evaluation of policy-based admission control mechanisms in NGN[C]//Proc of International Conference on Telecommunications. 2009: 137-142.

[8] MENTH M, LEHRIEDER F, BRISCOE B, *et al*. A survey of PCN-based admission control and flow termination[J]. IEEE Communications Surveys & Tutorials, 2010, 12(3): 357-375.

[9] 李靖波, 张光胜, 徐振阳, 等. 基于测量的自相似接纳控制算法研究[J]. 湖南大学学报, 2010, 28(12): 2819-2824.

[10] 齐俊杰, 胡洁, 麻信洛. 流媒体技术入门与提高[M]. 北京: 国防工业出版社, 2009.

[11] 李超, 张冬松, 金士尧. 面向流媒体应用的实时可执行概率获取方法研究[J]. 计算机应用研究, 2010, 27(9): 3462-3465.

[12] BRESLAU L, JAMIN S, SHENKER S. Comments on the performance of measurement-based admission control algorithms [C]//Proc of IEEE INFOCOM. Piscataway, NJ: IEEE Press, 2000.