

面向气象雷达数据传输的流媒体流量控制研究

蒋磊,冯径,王锦洲,沈晔

(中国人民解放军理工大学气象海洋学院,江苏 南京 211101)

[摘要] 气象雷达观测是气象观测的重要组成部分.气象雷达数据生成时间较长,且文件较大,在进行数据传输时,若采集完数据再传输会大大影响其数据的实时性要求.本文在分析雷达数据采集过程及特性的基础上,结合 RTP/RTCP 协议对拥塞判决控制进行改进,采用基于显式反馈的拥塞判决方案,设计了公平性的非线性流量控制方案,结合雷达数据生成特点,提出了使用流媒体技术的气象雷达数据传输算法 SM-RTA(Weather Radar Data Transmission Algorithm Using Streaming Media Technology).通过仿真实验证明,在 IP 网络中,SM-RTA 算法能为用户提供更稳定的实时性服务.

[关键词] 气象雷达数据,实时传输,RTP/RTCP 协议,拥塞判决,流量控制

[中图分类号] TP393.07 [文献标志码] A [文章编号] 1001-4616(2014)01-0081-06

Researches on Efficient Radar Data Transmission Control Mechanisms for Weather Service Networks

Jiang Lei, Feng Jing, Wang Jinzhou, Shen Ye

(College of Meteorology and Oceanography, PLA University of Science and Technology, Nanjing 211101, China)

Abstract: Weather radar observations play an important role in meteorological observation especially for short time forecasting. Weather radar generates large volume data and cost long time after getting completed scan file. So, it does not meet the requirements of real-time data merging in radar networking, which needs collection of multiple radar data at same time. The weather Radar data Transmission Algorithm using Streaming Media technology (SM-RTA) was proposed, which is based on radar data acquisition process and characteristics, combined with modified RTP/RTCP congestion judgment, and designed fairness nonlinear flow control program. Simulation results show that the SM-RTA can decrease data loss rate, delay and jitter to get more stable real-time services in IP network.

Key words: weather radar data, real-time transmission, RTP/RTCP algorithm, congestion judgment, flow control

随着现代科技的不断发展,雷达技术普遍应用于气象领域,气象雷达已成为天气预报和大气科学研究中不可或缺的工具.人们利用气象雷达探测大气中风、温度、湿度、气压等要素的时空分布,发现积雨云等降水系统,测定其空间分布、降水强度、降水量等,有些雷达还可测出晴空湍流、气象微粒的相态、形状.气象雷达为人们提供了大量丰富的信息.气象雷达观测是气象观测的重要组成部分,气象雷达数据资料的传输为短时、临近的天气预报等服务提供了可靠的保障.因此在雷达数据传输过程中,实时性是一项很重要的业务保障要求.一个好的数据传输算法能为气象业务提供更稳定、高效的服务.

目前流媒体传输广泛应用于网络中.流媒体传输对数据的实时性以及稳定性有着很高的要求,实时的视频与音频的播放为网络用户提供了更好的用户体验.在流媒体传输中,RTP/RTCP 算法得到了广泛应用.为了提高雷达数据传输中的实时性能,减少丢包率,提高传输质量,本文分析了雷达数据的采集传输流程,在此基础上结合流媒体数据传输 RTP/RTCP 算法,设计了新的流量控制方式以及雷达数据的发送接收流程,提出了 SM-RTA 算法.

收稿日期:2013-08-10.

基金项目:国防预研基金(407010401)、国家自然科学基金(61070174).

通讯联系人:蒋磊,博士研究生,研究方向:网络安全与管理. E-mail:326464320@qq.com

1 雷达数据的存储传输及数据格式

1.1 气象雷达数据传输流程

雷达的数据产生过程是:当雷达开始扫描时,本地磁盘设定的存储地址处即按照雷达数据命名规则产生相应时间与相应类型的雷达原始数据文件,文件中的数据随着雷达的扫描实时增加。

图 1 为雷达数据传输拓扑图,分为两个雷达组网、传输网和用户网 3 块. 雷达组网为中心节点提供雷达数据服务以及其他业务流服务. 用户网由中心节点及其他用户组成. 当雷达采集完数据后通过传输网传输给中心节点,中心节点处可能接收到的是雷达数据,也可能是其他业务流服务;雷达端通过 UDP 传输协议将雷达数据传输至中心节点并实时显示,中心节点将接受的数据按照用户的不同需要进行处理,分别传送给用户网中的其他用户^[1].

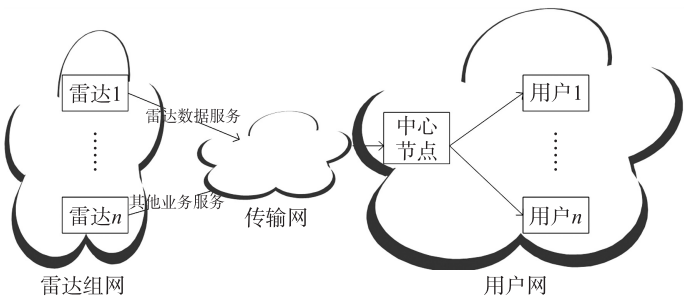


图 1 雷达数据传输拓扑图

Fig. 1 Radar data transmission topology

1.2 气象雷达数据格式

气象雷达的原始数据是指雷达回波经由信号处理分系统处理之后生成的、供实时处理软件用于提取信息和可视化显示的数据,主要包括文件头和文件内容两部分,文件头主要包含了雷达站名、站址、雷达型号、雷达主要参数、观测时间、扫描类型和雷达工作状况等信息. 气象雷达的原始数据可分为 3 类:体扫描(VCP)、平面位置扫描(PPI)和距离高度扫描(RHI)数据^[2]. 不同型号的雷达对数据的存储格式及名称设定不同。

以某型雷达为例,该种雷达提供了 3 种 VCP 扫描策略,如表 1 所示. 3 种策略分别扫描 5 ~ 14 圈,根据扫描方式的选择产生 20 ~ 70 Mb 的数据, PPI 与 RHI 数据约为 5 ~ 10 Mb^[3].

为提高中心节点实时显示效率,利用流媒体传输策略,一般使用“流水线”的方式,通过分层传输数据,节省数据产生和接收时间. 当第一层数据生成结束后就开始传输,因此总的时延大约为一层数据扫描时间,通常为 21 ~ 120 s. 由于 TCP 协议具有重传机制,能够提供更为可靠的数据传输,当数据扫描结束后,若有需要进行二次处理以提供给用户,可再利用 TCP 协议将雷达原始数据完整地传输给中心节点。

表 1 体扫描策略

Table 1 VCP Scan Strategy

体扫描策略	扫描模式	平均每层扫描时间/s
策略 1	5 min 内扫描 14 层	21
策略 2	6 min 内扫描 9 层	40
策略 3	10 min 内扫描 5 层	120

2 流媒体传输协议 RTP/RTCP

RTP 是目前常用的多媒体数据的实时传输协议,通常是基于 UDP 协议^[4]. 在流媒体数据传输中,音频与视频的传输对于实时性要求很高,但可容忍一定的丢包. RTP 协议基于 UDP 协议所设计,而 TCP 由于慢启动的需求,造成 RTP 协议相对于 TCP 协议开销小,带宽利用率大. 针对雷达数据的传输过程中,雷达扫描数据异地实时显示可以容忍一定的丢包,可利用 RTP/RTCP 协议进行数据的传输以提高传输效率. 但由于两节点之间并不仅仅有雷达数据的传输,有可能提供其他业务,此时 RTP 协议会尽可能地抢占带宽,以提高自身的传输速率,不限制发送数据,就会导致网络拥塞,对其他服务产生影响. 此时需要考虑的是数据传输的友好性,即在满足自身传输的速率要求,降低丢包的情况下,保证整条链路上所有服务的质量都得到保障。

在 RTP 会话期间,接收端会周期性地传送 RTCP 控制包,包中包含了已发送的数据包数、丢失的数据包数等统计资料,发送端根据这些信息来判断当前的网络拥塞情况,从而利用这些动态信息的改变调整传输速率^[5]. 传统的控制策略是使用加性增加、乘性减少(AIMD)的拥塞控制算法^[6]. 当网络出现拥塞时,使用乘法因子快速降低发送速率,当拥塞消除以后,使用加法因子来缓慢提高发送速率. 这种拥塞控制方式

属于基于丢包反馈的被动式拥塞控制机制. 研究表明,基于丢包反馈的控制机制由于使用丢包事件作为网络拥塞的判断依据,即便网络处于负载很高的情况下,只要没有产生丢包现象,协议就不会主动降低传送速率,因而最大程度地利用了网络带宽,提高了网络吞吐量. 然而在瓶颈链路状态下,这种协议表现出了很高的侵略性,一方面提高了网络速率,一方面也加大了网络丢包率的发生概率,造成整个网络的抖动性加剧. 在带宽竞争期,往返时延 RTT 较小的流相对于 RTT 较大的流,会获得更多的拥塞窗口增长次数,更容易抢占带宽.

根据这些缺陷,本文结合显式拥塞控制机制与基于往返时延流量控制机制的思想,在 RTP/RTCP 协议的基础上,运用 ECN 反馈机制与随机早期检测算法相结合对网络情况进行判决,在多条数据流在一条链路上传输的情况下,利用 TCP 的往返时延以及 RTCP 包返回时间作为公平性判定依据,引入公平性因子,提出基于公平性的非线性流量控制方案.

3 SM-RTA 算法设计

本文 SM-RTA 算法主要用于中心节点与用户之间的数据传输. 如图 2 所示,算法分为 4 个功能模块:中心节点数据录入模块、拥塞判决模块、流量控制模块、用户端数据接收模块. 中心节点端结合雷达数据特点,设计雷达数据录入模块,用于监测雷达数据生成及数据传输. 传输过程中通过拥塞判决模块、流量控制模块实时调整数据传送速率,以调高传输效率. 在用户端设计了数据接收模块.

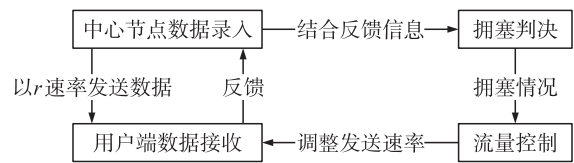


图 2 SM-RTA 算法功能模块

Fig. 2 The flow diagram of SM-RTA

3.1 基于显示拥塞通告及随机早期检测的拥塞判决

相对于基于丢包率的拥塞控制机制,基于显式反馈的拥塞控制机制通过中间节点反馈,能更为及时、准确地将拥塞信号传回发送端,避免了端到端协议中所产生的不及时和拥塞误判等问题.

显式拥塞通告(ECN)是一种常用的显示反馈形式的窗口管理机制,利用路由器检测自身的拥塞状态,直接发送反馈信息给发送端,发送端依据此拥塞信息调整发送速率及拥塞窗口^[7]. XCP、VCP 等协议都采用类似的显示反馈方式. 其中,VCP 运用了负载因子作为划分区间的标准. ECN 通过 IP 头中的两个 ECN 位标记拥塞状态,IP 头中的 ECN 位位于服务类型域(TOS)八位组的最后两位.

采用随机早期检测算法(RED)中的平均队列长度作为拥塞判据确定阈值,平均队列长度由指数滑动平均(EWMA)计算所得^[8]. 设置阈值 k_1 和 k_2 ,将队列缓存划分为空闲、轻度拥塞、阻塞 3 种状态,如图 3 所示. 结合 ECN 标示表示如表 2 所示,其中 $t(n)$ 表示第 n 个时隙下的队列长度, state 表示阻塞状态.

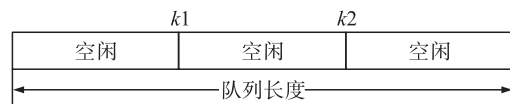


图 3 队列阈值设定

Fig. 3 Queue threshold setting

表 2 ECN 拥塞标示方式

Table 2 EcN Congestion Identification

队列长度	拥塞状况	ECN_1	ECN_2	State
$t(n) \leq k_1$	空闲	0	0	00
$k_1 < t(n) < k_2$	轻度拥塞	1	0	10
$t(n) \geq k_2$	阻塞	1	1	11
	不使用	0	1	01

3.2 公平性的非线性流量控制方式

通过上文分析,根据各个流 RTT 的不同,添加一个公平性因子,使得各个流在相同的时间内窗口趋近相等. 公平性因子的加入使得传输过程中不会出现一条流一直处于增长状态而占用过多的带宽,使得别的流传输速率下降.

假设网络中存在 N 条流进行数据传输,RTT 分别为 r_1, r_2, \dots, r_N , 对各流求加权平均数得 \bar{r} ,

$$\bar{r} = \frac{\sum_{i=1}^N r_i}{N}.$$

对于第 i 条流取公平性因子 f_i :

$$f_i = r_i / \bar{r}.$$

结合公平性因子与 MI/AI/MD 算法对拥塞控制方式进行改进:

$$x_i(n+1) = \begin{cases} x_i(n) * (1+\alpha) * f_i, & k1 \geq t(n); \\ x_i(n) * f_i + \beta, & k1 < t(n) < k2; \\ x_i(n) * f_i * r, & t(n) \geq k2. \end{cases}$$

式中, $x_i(n)$ 表示第 i 条业务流第 n 时隙发送速率; $t(n)$ 表示队列长度; $k1$ 和 $k2$ 为预先设定的阈值; f_i 为第 i 条业务流的公平性因子; $0 < \alpha < 1, 0 < \beta < 1$. 该式的含义为: 当网络处于空闲状态, 利用乘性因子快速增加发送速率; 当网络处于轻度拥塞状态, 利用加性因子缓慢增加发送速率; 当网络处于阻塞状态, 利用乘性因子快速降低发送速率; 每一种网络状况下, 利用公平性因子调节网络速率, 以达到惩罚抢占带宽行为, 激励新生流或速率较低流的目的.

3.3 中心节点数据录入及用户端数据接收方式

为提高数据传输的速率, 结合雷达数据的实时性, 考虑到文件生成速率可能会根据实际的情况有较大的波动变化, 若采用实时文件速率进行传输, 可能会对传输造成影响, 因此引入平均数据生成速率 r , 未传输文件大小 s 两个变量. 对雷达生成文件进行扫描, 设置两个指针 x, y , 其中 x 指针用于记文件生成大小 m , y 指针用于跟踪标记已发送文件大小 n . t 用于标定时间, 则可得到:

$$\begin{cases} r = \frac{m}{t}, \\ s = m - n. \end{cases}$$

以 r, s 两个变量进行数据传输速率的判决变量. 为保证雷达数据的完整性, 仍采用包传输的方式, 固定传送的报文大小, 当数据大小等于报文大小就将报文发送给中心节点. 设定指针标记体扫描每层数据文件的结束位置, 当每层数据扫描结束时, 即使小于设定的报文大小, 也将该段数据封包传送给接收端.

中心节点在硬盘中存储缓存文件, 实时录入接受的数据, 当一层扫描数据传输结束后, 即显示该层雷达图, 以达到实时观看的效果, 数据传送结束后, 中心节点即合并每层数据文件生成雷达数据文件.

3.4 SM-RTA 算法流程

结合拥塞判决及控制、数据录入存取设计, 设计 SM-RTA 算法如下:

雷达采集处监测雷达数据文件, 当产生新的雷达数据文件时, 进行标记监测. 先以 r 速率传输数据给中心节点, 设该数据流为到中心节点的第 i 条数据传输流. 中心节点通过 RTCP 反馈信息给雷达端, 雷达采集端根据接受的反馈信息进行拥塞判决. 当网络处于空闲状况 ($t(n) \leq k1$) 时, 根据雷达数据文件设定的两个指针判定剩余未传输的数据大小, 若 $s > 0$, 则以 $x_i(n) * (1+\alpha) * f_i$ 的速率进行传输, 若 $s = 0$, 则以 $x_i(n+1) = x_i(n)$ 的速率进行数据传输. 当网络处于轻度拥塞的状况 ($k1 < t(n) < k2$) 时, 以 $x_i(n+1) = x_i(n) * f_i + \beta$ 的速率进行传输. 当网络处于重度拥塞 ($t(n) \geq k2$) 时, 以速率 $x_i(n+1) = x_i(n) * f_i * r$ 进行传输. 中心节点接收数据并实时生成雷达图, 数据传输结束后生成雷达数据文件. 算法流程伪代码示例如下, 其中 out 为第 i 条数据流的传送速率, f_i 为第 i 条流的公平性因子:

```
Case:  $t(n) \leq k1$ :
    if ( $s > 0$ )
         $out = out * (1+\alpha) * f_i$ 
    else if ( $s = 0$ )
         $out = out$ 
    Endif
Endif
Case:  $k1 < t(n) < k2$ :
     $out = out * f_i + \beta$ 
Case:  $t(n) \geq k2$ :
     $out = out * f_i * r$ .
```

4 性能分析

数据传输中, 降低丢包率能够减少数据重传次数, 提高数据效率. 较低的传输时延能减少传输时间; 较小的延迟抖动提供稳定传输, 从而减少数据丢包; 提高吞吐量能够提升传输速率. 为了测试 SM-RTA 算法

性能,本文从丢包率、端到端延迟、延迟抖动和吞吐量4个方面分析算法性能。

利用 ns-2 网络模拟器分析本文算法的效果,在模拟实现中,结合雷达数据生成传输流程,设计拓扑结构如图4所示。

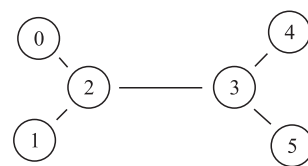


图4 仿真拓扑结构

Fig.4 Simulation of topology

节点2与节点3之间建立瓶颈链路,固定传输时延为20 ms。仿真实验分别做两次:第一次瓶颈链路接入普通流媒体算法,节点2绑定RTP协议,3绑定RTCP协议;第二次接入改进的算法,两侧接入链路带宽均为

10 Mbps,节点0与4之间建立TCP流,节点1与4之间建立TCP流,固定传输时延为5 ms,在TCP流上绑定FTP服务。两条FTP服务都在0 s时启动,1 s以后瓶颈链路数据流启动,共采样20 s。设定拥塞阈值 k_1 为平均队列长度的40%, k_2 为平均队列长度的80%。

4.1 丢包率

瓶颈链路带宽依次设为2、1.5、1 Mbps,仿真实验各做10次。比较RTP/RTCP算法以及采用改进算法数据流的平均丢包率。图5展示了在3种带宽环境下SM-RTA算法与RTP/RTCP算法平均丢包率的对比,SM-RTA算法的丢包率较RTP/RTCP算法的丢包率有了较大的减少。表3列出了在3种带宽环境下的丢失包数以及丢包率。上述数据显示,在带宽为2 Mbps时,链路状况较好,SM-RTA算法的丢包率仅为0.015,远远小于RTP/RTCP算法的丢包率;但是随着瓶颈链路的带宽减少,链路状况变差,当带宽为1.5 Mbps时,SM-RTA算法的丢包率数值提高为0.042,但是相较于RTP/RTCP算法,SM-RTA算法仍有着明显的优势;当带宽变为1 Mbps时,RTP/RTCP算法的丢包率显著上升,SM-RTA算法的丢包率仍维持在0.041。

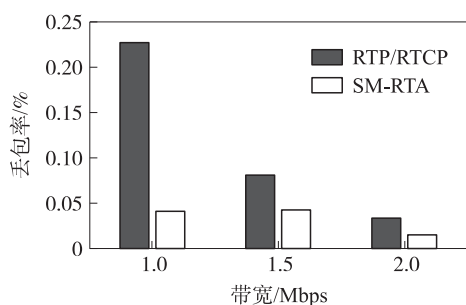


图5 不同带宽条件下两种算法平均丢包率对比

Fig.5 The comparison of two algorithms at different bandwidth condition

表3 平均丢包率对比

Table 3 The Comparison of Packet Loss Rate			
参数		RTP/RTCP	SM-RTA
2 Mbps	丢失报文数	169	75
	丢包率	0.033	0.015
1.5 Mbps	丢失报文数	410	211
	丢包率	0.081	0.042
1 Mbps	丢失报文数	1 155	209
	丢包率	0.227	0.041

4.2 端到端时延及延迟抖动

数据传输中,新的业务流的加入会导致原有业务流传输速率出现波动,增加了延迟抖动和数据丢包率,影响数据传输质量。SM-RTA中设计了公平性因子以提高算法容忍度,设计性能分析仿真如下:在瓶颈链路为2 Mbps的网络环境下,改变业务流启动顺序,0 s处启动节点0与节点5之间的业务流以及节点1到节点3的TCP流,3 s启动节点2到节点4的TCP流,仿真时间为20 s,截取前5 s端到端延迟数据,如图6所示。在0 s与3 s处,由于有TCP流一起运行,RTP/RTCP算法与SM-RTA算法端到端时延值都有波动。

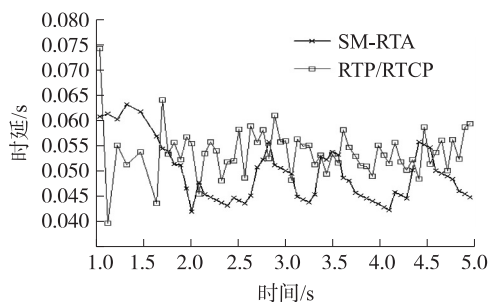


图6 端到端延迟

Fig.6 End-to-end delay

表4 时延均值及延迟抖动

Table 4 The average time delay and delay jitter		
参数	端到端时延均值/ms	延迟抖动
SM-RTA	49	4.6
RTP/RTCP	54	5.7

延迟抖动反应了算法的稳定性,其计算方法为时延的方差。表4中列出了两种算法的延迟抖动,可以看出SM-RTA算法的波动小于RTP/RTCP算法,故SM-RTA算法对于新业务流加入的容忍度好于RTP/RTCP算法。表4同时给出了SM-RTA算法与RTP/RTCP算法端到端时延均值,

SM-RTA 端到端延迟均值降低了 9%, 延迟抖动降低了 19%, 可见该算法具有更好的实时性与稳定性.

4.3 吞吐量

为比较 SM-RTA 算法与 RTP/RTCP 算法的吞吐量, 设计仿真实验: 瓶颈带宽设置为 2 Mbps, 节点 0 到节点 4 之间运行一条 TCP 流, 节点 1 到节点 4 之间运行一条 TCP 流, 两条 TCP 流在 1 s 启动. 节点 0 到节点 5 分别运行 RTP/RTCP 流与 SM-RTA 算法流, 仿真时间为 20 s, 取前 5 s 得到吞吐量, 如图 7 所示.

根据吞吐量值计算得到两种算法吞吐量均值如表 5 所示. 对比两种算法吞吐量均值得出, SM-RTA 算法比 RTP/RTCP 算法的吞吐量提高了 3.5%, 能很好地提高数据的传输效率, 为用户提供更流畅的实时服务.

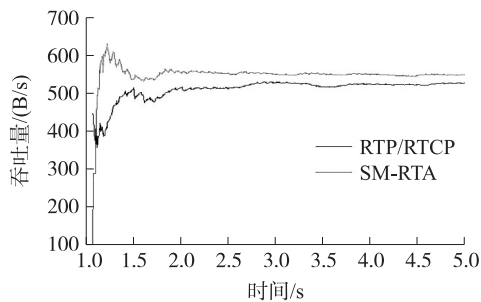


图 7 SM-RTA 与 RTP/RTCP 算法吞吐量

Fig. 7 Throughput of SM-RTA and Original RTP/RTCP

表 5 SM-RTA 与 RTP/RTCP 算法吞吐量均值

Table 5 The average throughput of SM-RTA and RTP/RTCP

参数	吞吐量均值/(B/s)
RTP/RTCP	526.54
SM-RTA	544.79

5 结语

本文通过分析雷达数据采集传输流程, 并基于流媒体技术设计了 SM-RTA 雷达数据传输算法. 通过仿真实验可以看出, 该算法在丢包率、延迟等 QoS 性能上均有一定优化, 公平性因子的引入也提高了 SM-RTA 算法在多条业务流并行情况下的容忍度. 但本文算法只考虑了基于点对点的传输, 并未考虑基于组播技术传输的可靠性, 在未来的工作中, 会对此进行研究.

[参考文献]

- [1] 王红艳, 刘黎平, 王改利, 等. 多普勒天气雷达三维数字组网系统开发及应用[J]. 应用气象学报, 2009, 20(2): 214-224.
- [2] 李忱, 张越, 陈赢. 气象雷达组网和网络气象雷达的发展[C]//第 26 届中国气象学会年会第三届气象综合探测技术研讨会分会场论文集. 杭州: 中国气象学会, 2009: 588-592.
- [3] 刘应军, 顾松山, 周雨华, 等. 新一代天气雷达体扫模式的比较分析[J]. 气象, 2006, 32(1): 44-50.
- [4] Luis Miguel A, Patricio A. Scalable streaming of JPEG 2000 live video using RTP over UDP[C]//International Symposium on Distributed Computing and Artificial Intelligence 2008(DCAI 2008). Spain: Springer, 2009, 50: 574-581.
- [5] Fernando Boronat Segui, Juan Carlos Guerri Cebollada, Jaime Lloret Mauri. An RTP/RTCP based approach for multimedia group and inter-stream synchronization[J]. Multimedia Tools and Applications, 2008, 40(2): 285-319.
- [6] Czerniak O, Altman E, Yechiali U. Orchestrating parallel TCP connections: cyclic and probabilistic polling policies[J]. Performance Evaluation, 2012, 69(3/4): 150-163.
- [7] Peter Jacko, Brunilde Sanso. Optimal anticipative congestion control of flows with time-varying input stream[J]. Performance Evaluation, 2012, 69(2): 86-101.
- [8] Abbasov B, Korukoglu S. Effective RED: an algorithm to improve RED's performance by reducing packet loss rate[J]. Journal of Network and Computer Applications, 2008, 32(3): 703-709.

[责任编辑: 严海琳]