

# 一种基于 gCHOKe 公平性的主动队列管理算法

黄亮亮,周井泉,李 琴

(南京邮电大学 电子科学与工程学院,江苏 南京 210003)

**摘 要:**文中主要研究了主动队列管理(AQM)的公平性算法。利用 gCHOKe 的多次击中能近似识别并惩罚非响应流,提高 gCHOKe 击中的有效性以及惩罚非响应流的力度,是提高算法公平性的关键因素。在 gCHOKe 算法的基础上提出了一种基于预处理的 pgCHOKe(preprocess based-gCHOKe)公平性算法,通过对数据流进行预处理筛选后再进行 gCHOKe 击中,提高了击中的有效性。仿真实验表明 pgCHOKe 相比 RED、gCHOKe 和 BLUE,可以获得更低的 UDP 吞吐量,是有效的、公平的。

**关键词:**拥塞控制;主动队列管理;BLUE;pgCHOKe

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2014)03-0063-04

doi:10.3969/j.issn.1673-629X.2014.03.016

## An Active Queue Management Algorithm Based on gCHOKe Fair

HUANG Liang-liang,ZHOU Jing-quan,LI Qin

(College of Electronic Science and Engineering,Nanjing University of

Posts and Telecommunications,Nanjing 210003,China)

**Abstract:**In this paper,study the fairness of the active queue management algorithm. The gCHOKe hit more than once improves the effectiveness and the force of punishing non-response flows are the crucial factors to enhance the fairness of algorithm,is used to approximately identify and punish non-response flows. Based on the gCHOKe,a new active queue management algorithm,pgCHOKe (preprocess based-gCHOKe), which based on the preprocessing has been proposed. The algorithm improves effectiveness of gCHOKe hit through using preprocess firstly, then the pgCHOKe hit. Simulation results show that compared with RED,gCHOKe and BLUE,pgCHOKe is able to get lower throughput,so it is effective and fair.

**Key words:**congestion control;active queue management;BLUE;pgCHOKe

## 0 引言

在现代通信和计算机网络中主要采用基于统计式多路复用技术的分组交换<sup>[1]</sup>,该方法可以更有效地提高网络的链路利用率<sup>[1]</sup>,然而当网络中数据包的数目持续增加时,将会慢慢地导致网络拥塞,此时网络资源将变得不能达到用户的需求<sup>[1]</sup>。故引进一个网络拥塞避免机制是必不可少的。特别是进入了大数据流时代,多媒体业务的逐步增加,各种不同性质的流在网络中也逐步出现,现代网络中的流类型主要可以分为适应性流和非适应性流<sup>[1]</sup>两大类。当网络拥塞发生时,非适应性流可以通过不段加大源端的发送速率来逐步抢占几乎所有的网络带宽资源,这将使得采用了源端拥塞避免机制的适应性流无法共享网络资源,因此,只

基于源端节点的拥塞避免机制<sup>[1]</sup>远远不能满足用户的需求。

现代网络中大部分拥塞控制算法都是基于中间设备的,网络中的路由器等中间设备检测链路的拥塞发生情况,当拥塞发生时,将拥塞情况反馈回去,并采取预防措施预防拥塞,关于这方面的研究主要是针对路由器的缓冲队列提出了一系列拥塞控制算法。当前的路由器队列管理算法主要有两大类:被动队列管理算法(PQM)和主动队列管理算法(AQM)<sup>[2]</sup>。被动队列管理算法在队列满时自动丢弃数据包,存在全局同步、死锁、满队列<sup>[3]</sup>等问题,而主动队列管理算法能有效监控队列长度的变化,并且能相对全面地审视各流带宽的公平性<sup>[3]</sup>以及对网络拥塞的影响,因此能够有效地控

收稿日期:2013-05-29

修回日期:2013-09-03

网络出版时间:2014-01-07

基金项目:江苏省普通高校研究生科研创新基金(CXLX12\_0471)

作者简介:黄亮亮(1987-),男,江苏常州人,硕士研究生,研究方向为网络的可靠性技术;周井泉,博士,教授,硕士生导师,研究方向为通信网络的信息管理和控制。

网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20140107.1650.031.html

制网络拥塞。

## 1 研究现状

第一个提出的 AQM 算法是随机早期检测 (RED) 算法<sup>[4]</sup>,也是目前主流的 AQM 算法。RED 算法的基本思想是路由器通过检测到达队列的平均队列长度来检测网络拥塞,一旦察觉拥塞将要发生,就利用概率判定机制随机地选择丢弃分组,减小平均队列长度,从而有效避免网络拥塞的发生和全局同步现象。RED 算法和传统的 PQM 尾丢弃算法相比较在避免持续满队列、死锁、全局同步等方面得到了改进,但是 RED 算法没有服务区分的能力,是尽力而为交付服务模型,没有考虑到各数据流和优先级之间的差异,使得非适应性流能够通过增大发送速率来抢占几乎所有的网络带宽资源,使适应性流在网络中几乎无法传输,使网络资源的整体利用率大大降低,网络公平性也受到了影响。

BLUE 算法<sup>[4]</sup>是 Feng 等人指出了 RED 的一些缺点,如参数设置困难及对队列长度过度依赖,队列长度不能很好地反映出网络的拥塞程度,因此拥塞通知信息的发送具有突发性而导致过多的丢包等等原因提出来的。该算法的基本思想是记录过去的丢包率和链路利用状态,以此达到进行调节丢包率  $P$  的大小的目的,因此该算法的自适应性比较强,但是仍然存在不少缺陷,比如它的参数设置仍然比较繁琐,另外,它对流的类型没有区分,如在某一类流的延迟时间比较长的情况下,仍然会很容易导致网络拥塞。

gCHOKe (Geometric-CHOKe) 算法<sup>[5]</sup>是一种基于 CHOKe 改进的算法,是一种完全无状态的近似公平的队列管理算法。在 gCHOKe 算法中,当一个数据包到达拥塞队列时,该算法将随机地从队列中抽取一个数据包与到达的数据包做比较,若两个数据包相同,则继续从队列中抽取一个数据包与之比较,若相同则继续抽取<sup>[5]</sup>,直达抽取的数据包与到达的数据包不相同为止,此时将到达的数据包与与之相同的数据包全部丢弃,并将抽取到的不相同的数据包重新放回队列中。进行 gCHOKe 击中后保留下来进入到队列中的数据包仍然要按 RED 算法进行丢包处理。

文中通过分析一些比较典型的主动队列管理算法的基本原理以及各自的优缺点,在 gCHOKe<sup>[5]</sup>的基础上提出了一种基于预处理的 pgCHOKe (preprocess based-gCHOKe) 算法。

该算法主要是根据非适应性流在网络中所占带宽多并且数据包数多的特点将到达的数据包先进行一次预处理筛选,然后将筛选过的数据包送入到 gCHOKe 算法中,来达到提高算法的公平性和适应能力的目的。

## 2 gCHOKe 算法分析

### 2.1 gCHOKe 算法原理

gCHOKe 算法的主要工作流程:当拥塞发生时,每次当一个数据包到达路由器,随机地从当前的缓存队列中抽取一个数据包与之比较,若该数据包与刚到达的数据包来自同一个流,则再从缓存队列中抽取一个数据包进行比较,直到抽取的数据包与刚到达的数据包不属于同一个数据流为止,此时将相同的所有数据包全部丢弃,并将不相同的那个数据包放回队列中。进入缓存队列中的数据包根据当时的拥塞水平通过 RED 算法计算丢弃概率,再按该概率对到达的数据包进行丢弃。当发生网络拥塞时,gCHOKe 算法对到达的分组采用的丢弃概率主要是由非适应性流的匹配程度影响,计算公式<sup>[5]</sup>为:

$$P_{\text{gCHOKe}} = \frac{2h - h^2}{1 - h} \quad (1)$$

式中, $P_{\text{gCHOKe}}$  为 gCHOKe 算法的击中概率; $h$  为非适应性流的匹配概率。

### 2.2 gCHOKe 算法存在的问题

gCHOKe 算法利用 gCHOKe 的击中概率进行丢包来达到区分适应性流和非适应性流的目的,当网络中流的数量较多时,惩罚的力度将会下降。

gCHOKe 算法击中的思想是根据非适应性流<sup>[6]</sup>占用带宽多且流的数目大的特点来进行非适应流的击中的,每来一个数据包都要进行  $n(n \geq 1)$  次击中<sup>[7]</sup>,并且每次都要将击中的所有包一起丢弃。当有多个相同的非适应性流接连到达时,第一个到达的数据包若与抽出的数据包匹配了  $n$  次且当第二个数据包与队列中抽出的数据包一次都不匹配时,非适应性流将会进入缓冲队列,后几个数据包的情况也相同,此时惩罚非适应流的效果将会大大降低。

## 3 pgCHOKe 算法

针对 gCHOKe 算法存在的惩罚力度不够的问题,在 gCHOKe 的基础上提出了一种基于预处理的 pgCHOKe 算法。

### 3.1 预处理机制的描述

pgCHOKe 算法在 gCHOKe 算法的基础上添加了过滤非适应性流的预处理机制,预处理机制是根据非适应性流在网络中数目多且占用带宽大的特点提出来的,特别是在大数据流占主导地位的现代网络中能够起到很大的作用。当网络发生拥塞时,通过对到达的数据包进行逐个对比,将出现重复次数大于等于两次的数据包进行丢弃,由于非适应性流数目多的特点,通过这种预处理机制能够很好地过滤掉非适应性流。带有预处理机制的 pgCHOKe 算法的工作情况如图 1 所

示。图中  $r$  为 RED 算法的丢包概率。

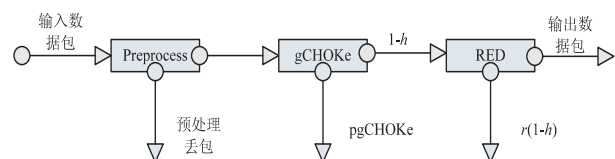


图1 带预处理机制的 pgCHOKe 算法框图

### 3.2 算法描述

pgCHOKe 算法的流程图如图 2 所示。具体工作步骤如下:

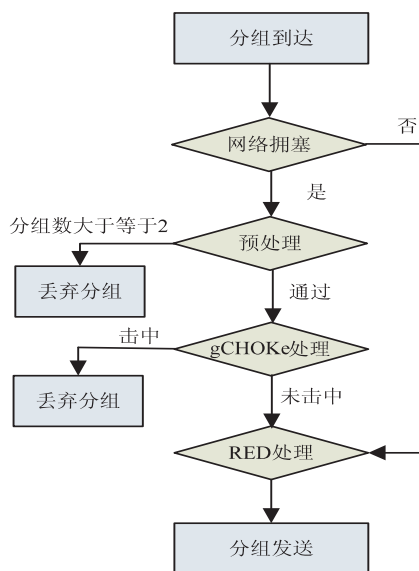


图2 pgCHOKe 算法工作流程图

步骤 1: 拥塞的判断阶段。

当分组到达后,采用 RED 算法拥塞判断机制来判断网络是否发生拥塞。若发生拥塞,进入步骤 2 预处理阶段;若拥塞未发生,进入步骤 4。

步骤 2: 预处理阶段。

进入预处理阶段以后,首先将到达的数据包储存在一张数据表中,并且记录下每个包的流标识(Stream ID, SID)<sup>[8]</sup>,然后将表中的数据包进行逐个比较(例如,首先将第一个数据包与 2, 3, ...,  $n$  ( $n$  为到达的数据包的数目)进行比较,若出现了大于或等于一个数据包与第一个数据包相同,则将这些相同的数据包全部丢弃。然后再从剩下的数据包中的第一个与 2, 3, ...,  $t$  ( $t$  为经过对比后剩余的数据包的数目 ( $t \leq n$ ))进行比较,丢包规则和前一次相同,如此往复,直到比较结束,然后将表中剩下的数据包逐个送入 gCHOKe 中,即进入步骤 3。

步骤 3: gCHOKe 阶段。

gCHOKe 模块首先将经过预处理后的数据包中第一个数据包与从缓冲队列中随机抽取的一个数据包进行比较,若两个数据包相同,即 gCHOKe 匹配,则继续从缓冲队列中随机抽取一个数据包与之比较,若相同,则继续从缓冲队列中抽取数据包进行比较,直到从缓

冲队列中抽取的数据包与到达的数据包不相同为止,此时将这些相同的数据包与到达的数据包一起丢弃。然后将没有 gCHOKe 匹配的数据包送入到 RED 队列中,即进入步骤 4。

步骤 4: RED 阶段。

该阶段依据 RED 算法来处理到达的分组。

### 3.3 算法复杂度分析

pgCHOKe 算法相对于 gCHOKe 算法来说多了一个预处理阶段,单从算法实现的角度来考虑,首先假设网络发生拥塞时数据表中存储了  $n$  个数据包,则当第一次比较完时数据表中的包的数目  $t$  一定小于等于  $n$ ,第二次比较时数据表中的数目一定小于等于  $t$ 。由此可知,当数据表中的数据包全部比较结束时,pgCHOKe 算法的时间复杂度<sup>[9]</sup>相对于 gCHOKe 而言,最坏的情况是出现在表中没有数据包两两相同的情况下,此时算法时间复杂度增加了  $n^2$  级,最好的情况是出现在数据表中的数据包全部相同,此时算法的时间复杂度得到了很大的优化。然而针对现代网络大数据流<sup>[10]</sup>的特点,每次到达的数据包经过预处理机制处理以后再送入 gCHOKe 机制进行处理的话,算法的复杂度肯定得到了很大的优化,因为只要有少量的数据包在预处理阶段被处理了对后面的 gCHOKe 机制来说将会大大减少算法处理的时间。

## 4 实验仿真与结果分析

为了使 pgCHOKe 算法的性能得到验证,在 NS-2.35 虚拟仿真平台上对 RED、BLUE、gCHOKe 和 pgCHOKe 算法分别进行实验仿真,根据仿真结果进行比较分析。实验网络拓扑结构如图 3 所示。各源端和路由器  $R_1$  之间的带宽为 10 Mbit/s,延时 1 ms;各目的端和路由器  $R_2$  之间的带宽为 10 Mbit/s,延时 1 ms;路由器  $R_1$  和  $R_2$  之间的带宽为 1 Mbit/s,延时 20 ms,形成了一条瓶颈链路<sup>[10]</sup>。

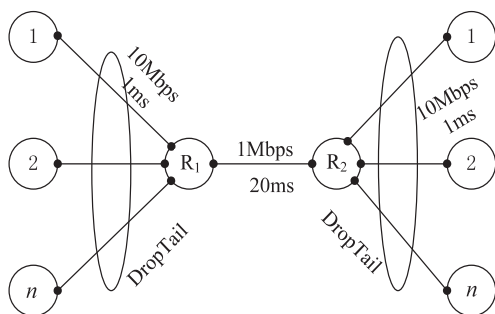


图3 实验仿真拓扑结构图

实验:

源端共有 92 个发送节点,其中 90 个节点发送稳定的 TCP 流,速率 0.1 Mbit/s;2 个节点发送 UDP 流,速率 0.2 Mbit/s;分组大小均为 0.5 kb,每个持续时间



为 115 s。其中一个发送 UDP 流的节点从 0 s 开始到 115 s 结束,另一个发送 UDP 流的节点从 5 s 开始到 115 s 结束,TCP 流的节点都是从 0 s 开始发送到 115 s 结束。

图 4 和图 5 分别显示了 RED、BLUE、gCHOKe 和 pgCHOKe 算法下 TCP 流和 UDP 流总的吞吐量的变化情况,由两图曲线可知 RED 算法在大量 UDP 流出现的情况下无法提供带宽公平性,网络带宽的绝大部分都被 UDP 流占据,TCP 流只能占用少部分的网络带宽;BLUE 算法相较于 RED 算法稍微改善了带宽公平性;由于 gCHOKe 算法提供了公平性调整机制,TCP 流和 UDP 流在网络中所占带宽的比例得到了很大的改善;相较于 RED、BLUE、gCHOKe 算法,带宽公平性方面做得最好的是 pgCHOKe 算法。

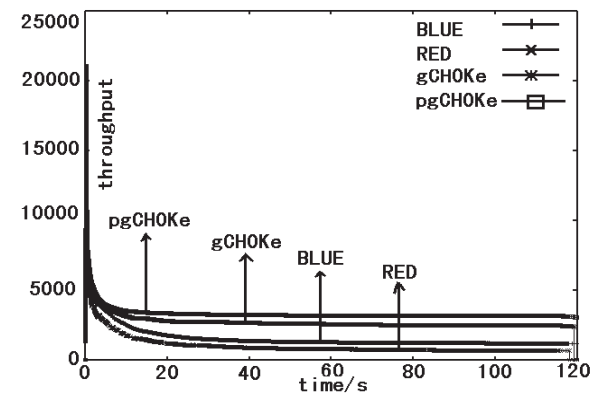


图 4 TCP 流吞吐量

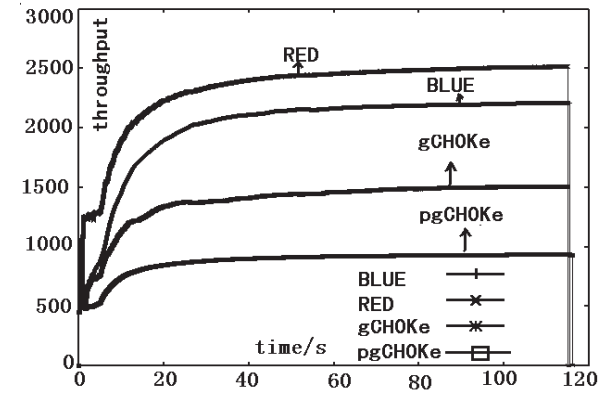


图 5 UDP 流吞吐量

表 1 显示在当前实验环境下 RED、BLUE、gCHOKe 和 pgCHOKe 算法发送的总数据包数目、UDP 丢失的数据包数、TCP 丢失的数据包数、UDP 的丢包概率以及 TCP 的丢包概率。由表 1 可以看出几种算法中 pgCHOKe 算法的 UDP 丢包概率最大,gCHOKe 算法次之,BLUE 算法第三,RED 算法最差。这也从侧面验证了 pgCHOKe 算法对 UDP 流的遏制能力最好,即其带宽公平性是最好的。同时 pgCHOKe 和 gCHOKe 算法的 TCP 丢包概率也较 RED 和 BLUE 算法要大,这也说明了 pgCHOKe 和 gCHOKe 算法也存在一定的误判<sup>[9]</sup>。

表 1 RED、BLUE、gCHOKe 和 pgCHOKe 算法的丢包概率

算法类型	发送的总包数	丢弃的 UDP 包数	丢弃的 TCP 包数	UDP 丢包概率	TCP 丢包概率
RED	187 908	3 783	2 097	0.020 132	0.011 16
BLUE	185 732	14 094	2 815	0.075 884	0.015 156
gCHOKe	182 204	38 423	5 930	0.210 897	0.032 546
pgCHOKe	170 114	57 884	3 479	0.340 27	0.020 45

5 结束语

通过对 gCHOKe 算法的性能分析,在其基础上提出了一种改进的 pgCHOKe 算法,在 gCHOKe 算法中加入一种非适应性流的预选机制,在数据包进入缓冲队列之前先对到来的数据包用预选算法进行初步筛选<sup>[11]</sup>非适应性并将其丢弃,然后再将经过筛选的数据包送入到 gCHOKe 算法中,在保护适应性流的基础上提高了对非适应性流的控制,对网络资源有了更好的分配,保证了各数据流的传输公平性。最后在 NS-2.35 仿真平台上对算法进行了仿真实验分析,仿真结果很好地证明了 pgCHOKe 算法比较有效地保证了网络带宽的公平性。但是改进的 pgCHOKe 算法由于仍然是建立在对到达的数据包进行随机比较以及后续的 gCHOKe 机制也是随机抽取数据包进行对比的基础上,所以对适应性流存在误判,故下一步研究的重点将放在如何减小算法对适应性流的误判<sup>[12]</sup>上。

参考文献:

[1] 田 泉,武 斌. ML-XCHOKe: 基于恶意度的改进 X-CHOKe 主动队列管理算法[C]//第九届中国通信学会学术年会论文集. 出版地不详;出版者不详,2012:44-48.

[2] 蔡文郁,张 昱,金心宇,等. RF-RED: 一种速率公平的 RED 改进算法[J]. 浙江大学学报(工学版),2007,41(4): 634-638.

[3] 姜 明,边 浩,陈 勤. HCHOKe: 改进的公平主动队列管理算法[J]. 计算机工程,2010,36(10):115-117.

[4] 刘 艺,孙鹤旭. 基于 BLUE 的主动队列管理算法分析[J]. 沈阳工程学院学报(自然科学版),2008,4(1):56-59.

[5] Tang Ao, Wang Jiantao, Low S H. Understanding CHOKe [C]//Proc of SIGMETRICS'03. San Diego, California, USA: [s. n. ],2003.

[6] Eshete A, Jiang Yuming. Protection from unresponsive flows with geometric CHOKe[C]//Proc of ISCC. [s. l. ]:[s. n. ], 2012.

[7] 聂 敏. 主动队列管理算法:CHOKe 性能分析[J]. 铜仁学院学报,2011,13(3):132-135.

[8] 姜 明,边 浩,张少丽. 改进的基于 CHOKe 击中历史的公平主动式队列管理[J]. 计算机应用,2010,30(2):289-291.

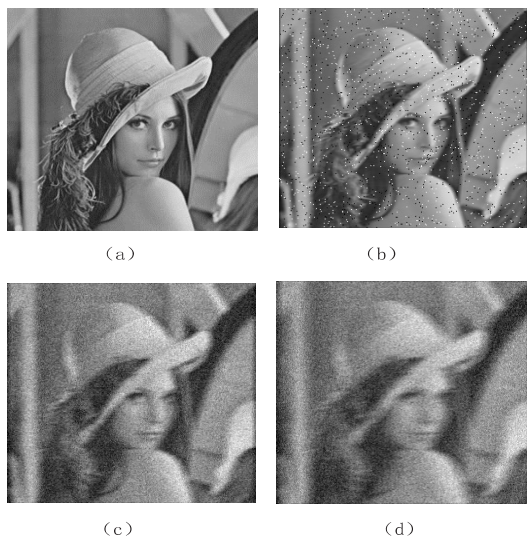


图 3 原图和加噪后模糊图像

从表 3 的评价数据可以看出, HSSIM 算法在对模糊图像进行质量评价时具有良好的抗噪能力, 得到的结果准确性高, 与主观评价基本保持一致。

## 4 结束语

在 SSIM 的基础上, 文中提出了一种新的图像质量评价方法 HSSIM。HSSIM 将直方图集中度所表示的图像模糊度应用到 SSIM 方法中, 对基于结构相似度的图像质量评价方法进行改进, 弥补了 SSIM 对严重失真图像和交叉失真类型评价不准确的不足, 并且具有较强的抗噪能力。实验结果表明, HSSIM 能更加准确地描述自然图像的质量特征, 同时也比较全面地考虑了影响图像质量的各种因素, 评价结果优于 MSE、PSNR 以及 SSIM, 且计算简单高效, 符合人眼视觉系统特性。

## 参考文献:

- [1] Mannos J L, Sakrison D J. The effects of a visual fidelity criterion on the encoding of images[J]. IEEE transactions on information theory, 1974, 20(4): 525-536.
- [2] Liang Yanling, Yang Chunling, Yu Yingling. Arbitrary shaped ROI image coding based on HVS[J]. Journal of South China university of technology, 2005, 33(3): 44-49.

- [3] Winkler S. A perceptual distortion metric for digital color images[C]//Proc of international conference on image processing. [s. l.]: IEEE, 1998: 399-403.
- [4] Wang Z, Bovik A C, Sheikh H R. Image quality assessment: From error visibility to structural similarity[J]. IEEE transactions on image processing, 2004, 13(4): 600-612.
- [5] Wang Zhou, Bovik A C, Lu L. Why is image quality assessment so difficult[C]//Proc of IEEE international conference on acoustics, speech, and signal processing. Orlando: [s. n.], 2002: 3313-3316.
- [6] Wang Z, Bovik A C. A universal image quality index[J]. IEEE signal processing letters, 2002, 9(3): 81-84.
- [7] Chou C H, Li Y C. A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile[J]. IEEE transactions on circuit and system for video technology, 1995, 5(6): 467-476.
- [8] 杨春玲, 陈冠豪, 谢胜利. 基于梯度信息的图像质量评判方法的研究[J]. 电子学报, 2007, 35(7): 1313-1317.
- [9] 李航, 路羊, 崔慧娟, 等. 基于频域的结构相似度的图像质量评价方法[J]. 清华大学学报(自然科学版), 2009, 49(4): 559-562.
- [10] 杨春玲, 高文瑞. 基于结构相似度的小波域图像质量评价方法的研究[J]. 电子学报, 2009, 37(4): 845-849.
- [11] Sampat M P, Wang Zhou, Gupta S, et al. Complex wavelet structural similarity: A new image similarity index[J]. IEEE transactions on image processing, 2009, 18(11): 2385-2401.
- [12] 任雪, 孙涵, 张金国. 一种新的基于局部特征的图像质量评价方法[J]. 中国图象图形学报, 2010, 15(8): 1236-1243.
- [13] Li Chaofeng, Bovik A C. Content-partitioned structural similarity index for image quality assessment[J]. Signal processing: Image communication, 2010, 25(7): 517-526.
- [14] 杨春玲, 徐小琳. 重视边缘区域的结构相似度图像质量评价[J]. 中国图象图形学报, 2011, 16(12): 2133-2139.
- [15] Firestone L, Cook K, Culp K, et al. Comparison of autofocus methods for automated microscopy[J]. Cytometry, 1991, 12(3): 195-206.
- [16] Sheikh H R, Wang Z, Cormack L, et al. LIVE image quality assessment database[DB/OL]. [2006-03-07]. <http://live.ece.utexas.edu/research/quality>.

(上接第 66 页)

- [9] 龚静, 吴春明. S-CHOKe: 一种增强 CHOKe 公平性的主动式队列管理算法[J]. 电子学报, 2010, 38(5): 1100-1104.
- [10] Govindaswamy V V, Zaruba G, Balasekaran G. Analyzing the accuracy of CHOKe hits, CHOKe misses and CHOKe-RED drops[C]//Proc of CCECE. [s. l.]: [s. n.], 2008.
- [11] Dow C R, Lin P J. A study of recent research trends and ex-

- perimental guidelines in mobile ad-hoc network[C]//Proceedings of the 19th international conference on advanced information networking and applications. Taipei, Taiwan, China: IEEE Computer Society, 2005.
- [12] Govindaswamy V V, Zaruba G, Balasekaran G. RECHOKe: A scheme for detection, control and punishment of malicious flows in IP networks[C]//Proceedings of IEEE GLOBECOM'07. Piscataway, NJ: IEEE Press, 2007: 16-21.

一种基于gCHOKe公平性的主动队列管理算法

作者：[黄亮亮](#)，[周井泉](#)，[李琴](#)，[HUANG Liang-liang](#)，[ZHOU Jing-quan](#)，[LI Qin](#)

作者单位：[南京邮电大学 电子科学与工程学院, 江苏 南京, 210003](#)

刊名：[计算机技术与发展](#)

英文刊名：[Computer Technology and Development](#)

ISTIC

年，卷(期)：2014(3)

本文链接：[http://d.wanfangdata.com.cn/Periodical\\_wjfz201403016.aspx](http://d.wanfangdata.com.cn/Periodical_wjfz201403016.aspx)