Computer Networks 91 (2015) 390–406

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

On fairness and application performance of active queue management in broadband cable networks

Gongbing Hong*, James Martin, James M. Westall

School of Computing, Clemson University, 100 McAdams Hall, Clemson, SC 29634, USA

ARTICLE INFO

Article history:
Received 5 March 2015
Revised 29 June 2015
Accepted 27 August 2015
Available online 7 September 2015

Keywords:
Cable access
Active queue management
Packet scheduling algorithms
Network performance evaluation

ABSTRACT

We evaluate the performance of modern delay-based active queue management (AQM) algorithms in downstream DOCSIS 3.0 cable environments. Our simulation-based study involves scenarios that include tiered service levels and application workloads that include FTP, HTTP-based adaptive streaming (HAS), VoIP, and web traffic.

Our results show that AQM maintains target queuing delays and consequently provides strong isolation between responsive high bandwidth and low bandwidth latency sensitive flows. We also show that lowering target queuing delays exacerbates TCP's RTT unfairness. Nevertheless, in the scenarios that we studied, observed application performance was not significantly impacted by the specific AQM in use.

With the potential large deployment of AQM imminent for DOCSIS 3.0, the analysis presented in this paper provides timely feedback to the community concerning how delay-based AQM can manage bandwidth allocation fairness and application performance in realistic downstream DOCSIS 3.0 cable network systems.

1. Introduction

The evolution of cable network technology is at an intriguing crossroad. Traditional broadcast video is converging with Internet video broadcast. Multiple system operators (MSOs) must engineer their access networks to competitively support both traditional video broadcast service and broadband Internet access. This task is challenging because of the rapid evolution of technology and customer demands from both worlds. In the video broadcast domain, system operators must provide access to a mix of hundreds of standard and high definition television channels along with video-on-demand services. In the broadband access domain, standards are rapidly evolving to provide ever increasing data rates to end users. To meet all requirements in current and emerging access networks, operators are rapidly migrating to a converged IP network for all cable and broadband access services.

Recent enhancements to cable access standards by CableLabs, the research and development organization for MSOs, are ensuring the convergence is technically possible. Current generation systems based on Data Over Cable Service Interface Specification (DOCSIS) version 3.0 (D3.0) can support multiple downstream, fixed bandwidth channels (usually 6.0 or 8.0 MHz) bonded together and multiple upstream channels (usually 6.4 MHz) bonded together[1]. A common configuration involving eight bonded downstream channels and four bonded upstream channels can support data rates up to 320 Mbits/s downstream and 120 Mbits/s upstream. Emerging systems will be based on the recently released DOCSIS 3.1 (D3.1) standard [2]. Higher channel bandwidth along with more efficient modulation and coding will allow downstream and upstream data rates of up to 10 and 1 Gbps, respectively.

* Corresponding author. Tel.: +1 6786369539.
E-mail addresses: hgb.bus@gmail.com, gongbih@clemson.edu (G. Hong), jmarty@clemson.edu (J. Martin), westall@clemson.edu (J.M. Westall).

http://dx.doi.org/10.1016/j.comnet.2015.08.018
1389-1286/

Scheduling and queue management disciplines are fundamental to computer networking and have been studied from many different perspectives. Common objectives include congestion management and provision of fair sharing or differentiated services in IP networks. In spite of this body of knowledge, access networks can still suffer from known problems including bufferbloat, TCP RTT unfairness, and vulnerability to unresponsive flows [3–8]. Shared medium cable networks present additional challenges. On one hand they must attempt to ensure that bandwidth sharing and latency objectives among many competing flows are met. On the other hand, regulatory constraints and concerns pertaining to so-called "net neutrality" limit the mechanisms that can be used in doing so. These regulatory concerns are so strong that flow-aware, multi-queue fair share scheduling systems such as deficit round robin (DRR) or self-clocked fair queuing (SCFQ) that would greatly facilitate bandwidth management are not currently in use.

Two flow-agnostic, single queue active queue management (AQM) algorithms that have been proposed since 2012 are Controlled Delay (CoDel) [4,9] and Proportional Integral Controller Enhanced (PIE) [10,11]. CoDel has been evaluated in different types of networks including 802.11 and high speed wired networks. PIE, which is not yet widely deployed, is based in part on concepts that were introduced previously [12]. The D3.1 standard requires that a cable modem (CM) manages upstream best effort traffic using PIE. Wide deployment of D3.1 equipment is not expected for several years. This prompted CableLabs to modify the current, and widely deployed, D3.0 standard to allow the immediate use of the PIE AQM for upstream traffic [1]. For downstream traffic, the Cable Modem Termination System (CMTS) must support a published AQM algorithm, not necessarily CoDel or PIE, in both D3.0 and D3.1. The standards indicate the equipment must support AQM by default, although a network operator can choose to override the default configuration, possibly selecting to not use AQM or to use an alternative method. The intent of adding the AQM requirement to the DOCSIS standards is to ensure that network operators have tools that can address bufferbloat in a consistent manner. With large scale deployments of delay-based AQM algorithms imminent, there is critical need to understand how these schemes will behave in current and emerging cable networks.

In prior work we have developed an *ns2*-based simulation model of DOCSIS 3.0 [13]. We have used the simulator to explore upstream buffer management in CMs [14]. The focus of that paper was upon preventing upstream bufferbloat in the CM. The validation of downstream scheduling disciplines in bonded channel environments was the focus of [15]. Extending our ongoing work in the area of AQM [16], the research presented in this paper provides detailed analysis on the effectiveness of recently proposed delay-based AQM schemes applied to downstream traffic in cable access networks. Our simulation framework utilizes realistic scenarios involving application traffic models such as FTP, VoIP, web browsing, and HTTP-based adaptive streaming (HAS). The analysis includes scenarios involving both single and bonded downstream channels. We consider scenarios that involve different service tiers, where different cable users are provisioned with different data rates.

To the best of our knowledge, our study is the first to assess the ability of modern delay-based AQM schemes to manage fairness and application performance in realistic, single or bonded channel downstream DOCSIS 3.0 cable network scenarios. The results presented in this paper address the following questions: (1) How effectively do CoDel and PIE support fairness and application performance in realistic cable network scenarios? (2) Are there undesirable side effects when the AQM interacts with tiered service levels? (3) How effectively do the schemes isolate responsive traffic from unresponsive flows?

This paper is organized as follows. Section 2 summarizes relevant research that has been published in the literature. Section 3 introduces the experimental methodology. The next three sections contain the results of our simulation studies with the focus of each section being one of the three fundamental questions enumerated above. Section 7 provides conclusions and identifies our next steps.

2. Related work

Bufferbloat has received significant attention recently [3,4]. Bufferbloat is a side-effect of congestion that occurs when network equipment is provisioned with large unmanaged buffer queues. Its effects are persistently high queue levels that in turn lead to large and sustained packet latency.

AQM has long been considered a solution to bufferbloat. The random early detection (RED) algorithm manages a queue by randomly dropping packets in a manner in which the random drop rate is dynamically adjusted based on an average queue size estimate and a configured maximum allowed drop rate (referred to as *maxp*) [7]. Most RED implementations offer the 'gentle' option where the drop rate increases linearly from *maxp* to 1 once the average queue level exceeds the target queue size [8,17].

While RED is widely available, it is not widely used. It has been shown that the average queue delay with RED is sensitive to traffic loads and to parameter settings[18,19]. Adaptive RED (ARED) is a simple extension to RED that further adapts the random drop process such that the average queue level tracks a target queue level [20]. This adaptation is performed periodically. We refer to this parameter as the *control_interval*. We refer the reader to [21] for a thorough summary of relevant AQM research.

CoDel and PIE are AQM algorithms that were specifically designed to address the shortcomings of (A)RED. Both AQMs are delay-based as they proactively drop packets to ensure average packet queue delay remains less than a configured latency target. We refer to this as the *target_delay* parameter. Both AQMs expose a second configuration parameter analogous to the *control_interval* of ARED that defines the timescale of control.

CoDel's delay estimate is based on a per packet latency monitor. PIE's delay estimate is based on an estimate of the recent departure rate at the queue. The two AQMs both tolerate infrequent bursts of traffic. However, the details of the burst control mechanisms differ and are described in [4] and [10], respectively.

The PIE algorithm performs early packet drops as packets arrive at the queue. The CoDel algorithm, as originally proposed in [4,9] performs early packet drops as queued packets