Journal of Computer and System Sciences 77 (2011) 898–916

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

Controlling mean queuing delay under multi-class bursty and correlated traffic

L.B. Lim [a,*], L. Guan [a,*], A. Grigg [b], I.W. Phillips [a], X.G. Wang [c], I.U. Awan [d,e]

[a] Department of Computer Science, Loughborough University, UK, LE11 3TU
[b] System Engineering Innovation Centre (SEIC), Loughborough University, UK, LE11 3TU
[c] Department of Computing and the Digital Environment, Coventry University, UK, CV1 5FB
[d] Department of Computing, University of Bradford, UK, BD7 1DP
[e] Visiting Faculty in the College of Computer & Information Sciences, King Saud University, Saudi Arabia

ARTICLE INFO

Article history:
Received 20 January 2010
Received in revised form 26 April 2010
Accepted 9 August 2010
Available online 17 August 2010

Keywords:
Discrete-time queuing model
Dynamic queue thresholds
Superposition of *N* MMBP-2
QoS
Closed-loop feedback control
Adaptive queue management

ABSTRACT

This paper presents an adaptive queue management scheme to maintain queuing delay in a router at a required level based on a comprehensive analytical model under aggregated Internet traffic flows from various traffic classes. The proposed scheme uses a closed-loop feedback control mechanism to constrain the average queuing delay by regulating traffic arrival rate implicitly through a movable queuing threshold. A discrete-time queuing model is developed to derive the relationship between average queuing delays and queuing thresholds based on a traffic model that models aggregated Internet traffic through superposition of *N* MMBP-2 arrival processes. The queuing threshold is adjusted dynamically with reference to the relationship derived in the analytical model and also feedback of average queuing delay measurement. Packets are dropped dynamically with respect to the changes of queuing threshold and the packet loss events serve as implicit congestion indicators. Matlab is used to perform queuing analysis and simulation. Statistical evaluation is performed to show the efficiency and accuracy of the analytical and simulation results.

1. Introduction

With the advances of networking technologies, various types of networks have been deployed and interconnected to form larger heterogeneous networks [1]. This is crucial to enable ubiquitous and pervasive communication in our daily life. Sharing of knowledge and communications are no longer hindered by distances and costs involved. The booming in Internet technology and networking infrastructures has led to more cost-effective and time-effective data sharing and communication in our daily life.

That said, the Quality of Service (QoS) perceived by users is a major concern for the success of next generation networks with growing demands of real-time communication to relay multimedia traffic, timing critical tele-operation control data and sensory data. Various QoS control approaches have been introduced in order to improve network performance and to fulfill the Service Level Agreement (SLA).

Each QoS control scheme has its own goals to be achieved in terms of timeliness of data delivery, system throughput, packet loss rate, etc. Timeliness of data delivery is one of the vital QoS performance measures. Delay in data delivery for real-time communication may annoy the users. In addition, data may be deemed useless or redundant if it arrives later than

* Corresponding authors.
E-mail addresses: L.B.Lim@lboro.ac.uk (L.B. Lim), L.Guan@lboro.ac.uk (L. Guan).

0022-0000/$ – see front matter
doi:10.1016/j.jcss.2010.08.007

the deadline. Lateness in delivering data may even cause system failures or disasters to happen. Therefore, network delay control is of high importance in QoS provisioning.

End-to-end network delay is an aggregation of nodal delay along the transmission path while nodal delay is a function of various delay components at a node. Nodal delay is comprised of stochastic delay components (propagation delay and transmission delay) and deterministic delay components (queuing delay and processing delay). Among the nodal delay components, queuing delay is easier to control. This is because the maximum queuing delay experienced by a system is determined by the queue size and also system utilization. However, it is not a trivial task to find an optimum queue size or threshold. Inappropriate setting of the queue size or threshold may lead to system performance degradation such as large queuing delay if the queue size is too large or high packet dropping rate if the queue size is too small [2].

There are a lot of Active Queue Management (AQM) schemes (e.g. Random Early Detection (RED) [3] and its variants [4–9], Proportional Integral (PI) controller [10], Random Early Marking (REM) [11], Loss and Queuing Delay (LQD) [12], BLUE [14], GREEN [16]) which run on routers have been introduced to alleviate congestion problems in the networks. RED [3] is the most well-known congestion avoidance mechanism for packet-switched networks. It becomes the core of design for most AQM schemes introduced since. Most current researchers use the same concept as RED to drop or mark packets between minimum and maximum thresholds but change the way of dropping packets or calculating dropping probability. For example, DSRED [7] maintains the same mechanism in RED but with a further threshold introduced in between the minimum and maximum thresholds as the changing point to change the steepness of the dropping function slope. Another RED variant proposed by Al-Raddady and Woodward [8] recently uses an adaptive dropping probability based on the target arrival rate and average arrival rate, instead of linear growing dropping probability in RED and AIMD in ARED [6]. CHOKe [13] is also based on RED. When the average queue length falls between the minimum and maximum thresholds, the packet will be dropped if it is from the same flow as the previous packet; otherwise the packet will be dropped based on the RED dropping probability. Feng et al. [14] have proposed an AQM scheme called BLUE; packet loss and link idle events are used to manage congestion instead of instantaneous queue length or average queue length. SFB [15] is a variant of BLUE algorithm which incorporate a BLOOM filter to identify unresponsive flows and then rate limit the unresponsive flows. GREEN [16] aims to provide fairness during congestion at edge router. The dropping probability is calculated based on the throughput of incoming traffic and number of active flows in the router.

In general, most of the AQM schemes introduced monitor network load in order to anticipate and avoid congestion inside the networks. Through these congestion avoidance mechanisms, average queuing delay at the routers can be lowered but yet not bounded per QoS requirement. Besides, some of the AQM schemes have faced some difficulty in parameter tuning, such as queue thresholds and dropping probability, to achieve an optimum system performance. An adaptive queuing threshold is indeed needed in order to bound the average queuing delay. Furthermore, traffic characteristics and system utilization are not taken into consideration explicitly for the AQM design. System performance may vary with different traffic loads and processing loads.

In this paper, a control theoretic queue management scheme for a router, referred to as Dynamic THreshold (DTH), has been developed. The proposed scheme combines a closed-loop feedback control mechanism together with a discrete-time analytical model derived for multi-class aggregated Internet traffic using N MMBP-2 to maintain average queuing delay in a router. The remainder of the paper is organized as follows: Section 2 discusses on related work; Section 3 gives an overview of DTH system; Section 4 describes the system model of DTH queue management scheme; Section 5 then presents the analysis from analytical and simulation results; and lastly, Section 6 gives some conclusions and suggestions for future work.

2. Related work

There are some control theoretic AQM schemes that aim to control the average queue length to its required target, but determining the optimum target queue length remains an unresolved issue. PI controller [10] aims to regulate the queue length to a required value using a closed-loop feedback control. The dropping probability is updated periodically based on the queue length. The same approach is taken by Hong et al. [17]; the proposed AQM scheme aims to control the average queue length via an adaptive dropping probability which is calculated based on the average queue length, target queue length and estimated packet arrival. LQD [12] is an AQM scheme that adapts the dropping probability based on target queue length and target loss rate. While REM [11] uses a cost function to determine the dropping probability based on rate mismatch observed between packet arrival and packet departure and queue length mismatch observed between actual queue length and target queue length. Despite the fact that these approaches may control the average queue length to its target level; the average queue length is bounded with the assumption of maximum system utilization. However, the same target queue length may be translated into different average queuing delay with different system utilization and also traffic characteristics. In conclusion, the average queuing delay in a router cannot be bounded using the existing approaches.

Apart from that, Internet traffic that traverses through routers is an aggregated traffic flow from various traffic classes and applications. The aggregated network traffic is bursty and also exhibits Long Range Dependence (LRD) characteristic [18,19]. LRD shows that the behavior of a time-dependent process is correlated over a range of time-scales. Internet traffic modeling has remained an open research issue over the years as there is currently no perfect model that can model the Internet traffic. Poisson process and Bernoulli process fail to model bursty and correlated traffic. Due to the simplicity

of the mathematical model and characteristics of Markovian arrival process, it has become a popular model for bursty and correlated traffic modeling. For example, Markov Modulated Bernoulli Process (MMBP-2) [20,21] and Markov Modulated Poisson Process (MMPP) [22,23] can be used to serve the purpose. It was shown that superposition of MMPP can be used to model variable packet traffic with LRD [22,23]. MMBP is the discrete-time counterpart of MMPP. Thus, superposition of multiple (N) MMBP is a good candidate for traffic arrival process to model aggregated Internet traffic in discrete-time queuing analysis. A discrete-time approach is used for the proposed scheme instead of a continuous-time approach as computers and communications are discrete in nature. Only a single event can occur at any time instant for a continuous-time approach; whereas a discrete-time approach can allow multiple events to take place [24]. Even though continuous-time approach is less complex, it is not able to represent the digitalized communication world well.

The mapping relationship between queuing threshold and average queuing delay that takes into account of traffic arrival process and system utilization are not well explored. The adaptive queuing threshold approach based on approximate aggregated Internet traffic modeled by multiple discrete-time Markovian source models is the key feature which differentiates the proposed scheme in this paper from previous work in this area to constraint the average queuing delay. Guan [25] has proposed a movable threshold queue management scheme with a single Markovian source; a single traffic source is not sufficient to model aggregated Internet traffic that consists of multi-class traffic. Even though Al-Jabber [26] and Wang [27] propose to use multi-source in establishing the delay maintaining mechanism, the arrival process used is based on Binomial distribution. A Binomial arrival process is not able to capture the burstiness and LRD characteristics of Internet traffic. Therefore, our approach extends the previous works by giving a comprehensive investigation on the discrete-time queuing model to use multiple MMBP-2 arrival processes in order to capture multi-class aggregated Internet traffic. This paper is an extended version to the paper [28] published previously with more system performance analysis such as throughput and packet loss.

3. DTH overview

The DTH mechanism relies on the relationship mapping between queuing thresholds and average queuing delays that are generated a priori for potential scenarios based on the targeted traffic profiles and system utilization. The traffic profiles and system utilization can be approximated from the past measurements collected by the service provider at the routers. The main aim of the offline queuing analysis model with on-line queuing threshold adjustment is to bound the average queuing delay to the required value and yet maximizing the system utilization.

DTH aims to control the average queuing delay at network routers to a required average queuing delay (TD) by adjusting the queuing threshold (L) dynamically as shown in Fig. 1 through a closed-loop feedback controller based on the relationship derived by the analytical model. The feedback control loop results in a movable queue threshold for the system queue and maintains the system average queuing delay around the TD specified.

The average queuing delay (D_k) is measured and calculated at a fixed time interval denoted as time window (TW) with associated index k ($k = 0, 1, 2, \ldots$). The D_k measured is used to obtain the delay delta (G_k) between the measured and target average queuing delay. The delay delta, G_k, is then used to predict the target average queuing delay ($\hat{D}_{k+1}$) for the next TW based on Eqs. (1)–(3) [25].

$$\begin{aligned} G_k &= kTD - \sum_{i=1}^{k} D_i; \quad k = 1, 2, \ldots \\ &= TD - D_k + G_{k-1} \end{aligned} \tag{1}$$

$$TD = \frac{D_1 + D_2 + \cdots + D_k + \hat{D}_{k+1}}{k+1} \tag{2}$$

$$\hat{D}_{k+1} = 2 \times TD - D_k + G_{k-1}; \quad k = 1, 2, \ldots \ \& \ G_0 = 0 \tag{3}$$

After obtaining the $\hat{D}_{k+1}$, the $\hat{D}_{k+1}$ is then being translated into a queue threshold, L, based on the mapping relationship obtained from the queuing analysis in Section 4. The queuing threshold is either increased, if a lower queuing delay is observed in the previous time step to allow larger queuing delay in the next time step, or decreased to constrain the queuing delay. With the DTH queue management scheme, packets are dropped when the current queue length ($QLen$) exceeds the target queue threshold obtained from the mapping table. Packet loss events can serve as implicit congestion indicators for the sources to regulate their sending rates in order to avoid overloading the routers and also avoid a high packet loss rate.

4. DTH system model

From the DTH mechanism explained in Section 3, in order to maintain average queuing delay at its target level, the mapping relationship between queuing delay and queuing threshold is very important apart from next target average queuing delay estimation. A discrete-time queuing model is used to generate the mapping relationship with the assumption of packet departure probability (service rate) and packets arrival probability (arrival rate) are known. The queuing model introduced here uses N MMBP-2 with $N > 0 \in \mathbb{Z}$ to represent aggregated Internet traffic.

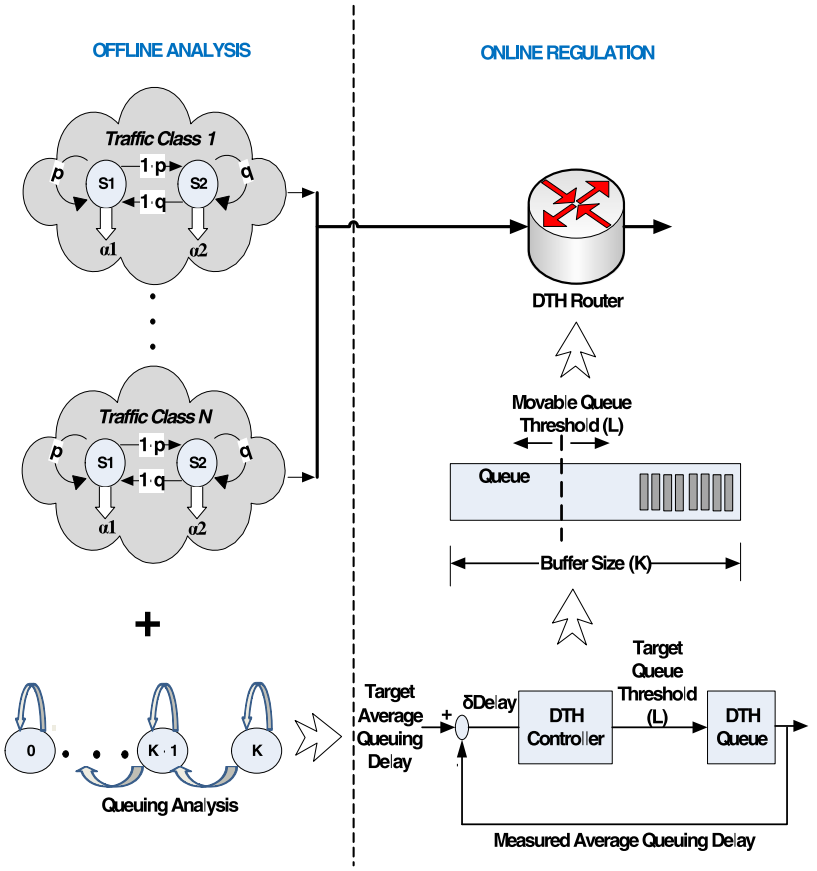


Fig. 1. DTH system diagram.

The queuing analysis is carried out by varying the queue threshold with superposition of N MMBP-2 sources. The queuing analysis with the same arrival process and the same departure probability/service rate (β) is carried out for all queue thresholds, which ranges from 1 to K (maximum queue size). Average queuing delays observed for each queuing thresholds are then derived through the steady state queue length probabilities obtained. After K iterative of queuing analysis; a mapping relationship between the threshold and delay is obtained from the analytical results.

With the DTH controller and also periodic feedback of average queuing delay measurement; DTH buffer management controller can then adjust the queue threshold for next time unit to maintain the average queuing delay to the required value. Following subsections further discuss on the traffic model and also queuing model for this buffer management scheme.

4.1. MMBP-2 traffic model

MMBP-2 source model (Fig. 2) is used as a traffic source model to represent bursty traffic [20,21,25]. There are two distinct states in a MMBP-2 arrival process, which are state $S0$ and $S1$. The MMBP-2 source model is characterized by a transition probability matrix $\tilde{P}$ and a diagonal arrival probability matrix $\tilde{\Lambda}$ as given in Eqs. (4) and (5).

$$\tilde{P} = \begin{bmatrix} p & 1-p \\ 1-q & q \end{bmatrix} \tag{4}$$

$$\tilde{\Lambda} = \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix} \tag{5}$$

The MMBP-2 source can be used to model traffic flow with different characteristics by setting the state transition parameters (p, q) and arrival probabilities (α_1, α_2) of the model appropriately. It can become a Bernoulli source by setting the same arrival probability of both states in MMBP-2 to generate smooth and constant traffic patterns. It can also be used to model voice or video traffic. By setting arrival probability in either $S0$ or $S1$ to zero, MMBP-2 source becomes an IBP [30] source which has ON and OFF state. OFF state represent the silence state in the voice traffic. In short, MMBP-2 source can

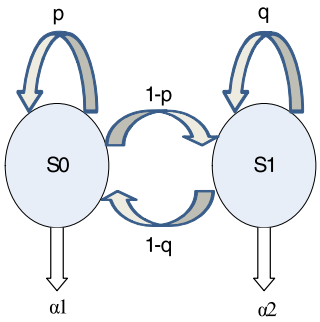


Fig. 2. MMBP-2 diagram.

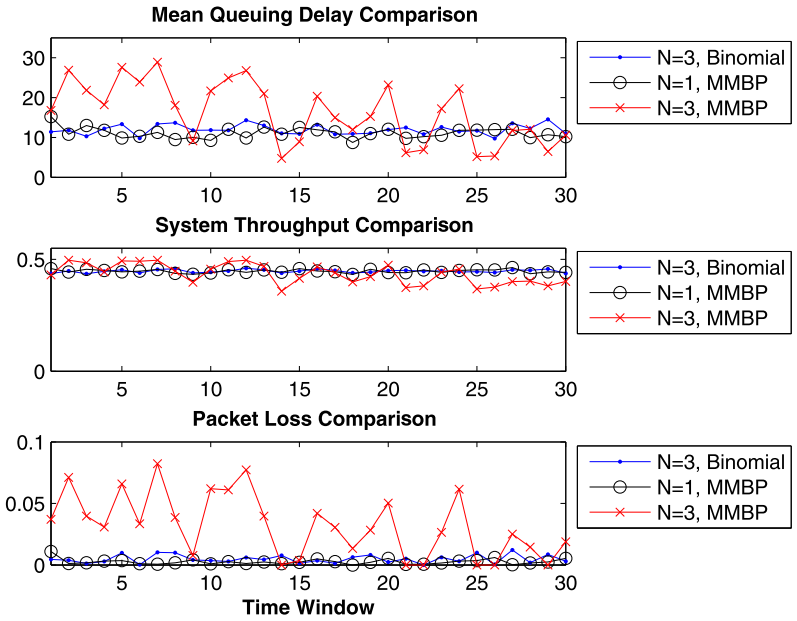


Fig. 3. System performance with different arrival process.

be adjusted to generate different traffic patterns with different traffic intensity or burstiness for traffic flows from different traffic classes.

Nevertheless, a single MMBP-2 arrival process is not able to represent multi-class aggregated Internet traffic. Internet traffic in the core network is an aggregation of multiple traffic flows from different traffic classes, such as CBR (Constant Bit Rate), VBR (Variable Bit Rate), voice, video, etc. The queuing model with a single traffic source is not able to model the aggregation of traffic flows. Therefore, superposition of multiple MMBP-2 models is proposed instead to model the aggregated traffic flows at core networks.

A simple simulation using Drop Tail queue management ($queue\ size = 20$) with different arrival process has been carried out to show why superposition of N MMBP-2 is chosen as the sources for DTH model instead of single MMBP-2 [25] or Binomial sources [27]. From Fig. 3, it can be observed that the mean queuing delays measured from the simulation for single MMBP-2 traffic source and Binomial traffic sources shows the similar trend with smooth average queuing delay. This implies that single MMBP-2 or Binomial sources are not able to capture the burstiness of multi-class Internet traffic and thus may not give an accurate offline queuing analysis for DTH. Besides the mean queuing delay, the other system performance metrics, such as throughput and packet loss, would fluctuate with the burstiness of the traffic also.

4.2. Queuing model

A discrete-time queuing model is adopted for offline analysis of the proposed queue management scheme. The model proposed is a N-MMBP-2/Geo/1/K queue follows the Kendall's notation. In a discrete-time queuing model, time is divided into slots of fix length interval. Inside a time slot, a packet arrival or/and departure may take place with the departure

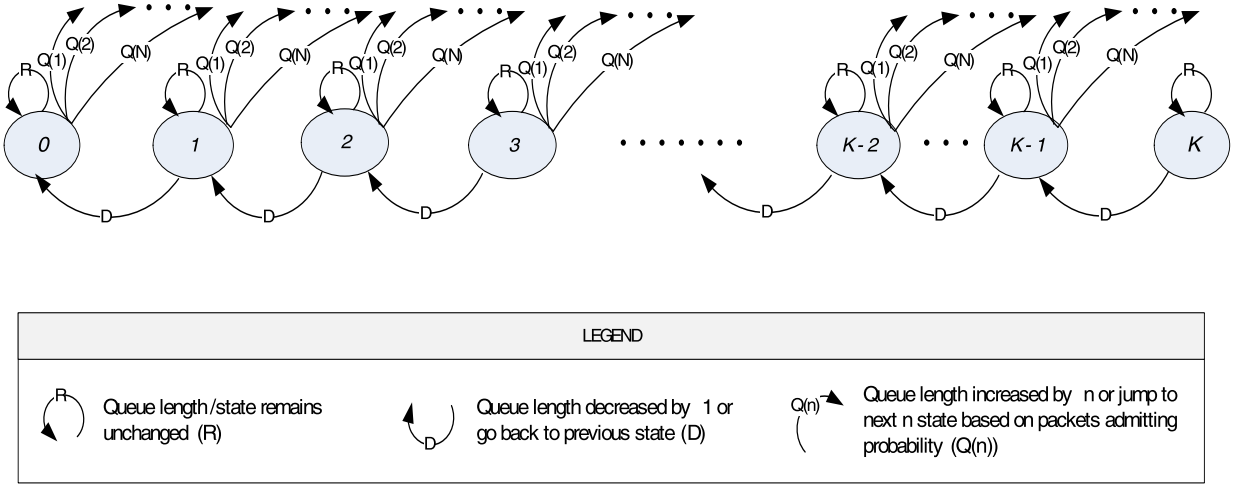


Fig. 4. State transition diagram for superposition of N MMBP-2 arrival process in single dimension.

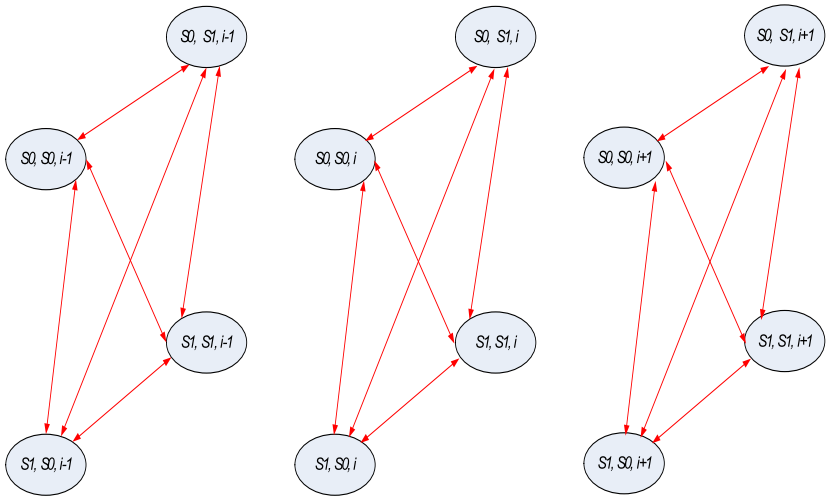


Fig. 5. R elements in $\tilde{QT}$.

probability of β. For the queuing model proposed, a packet departure is assumed to take place before arrival in any time slots and at maximum one packet departure is allowed. The queue size is finite for the model proposed, which means the queue can hold up to maximum K packets (where $K > 0$). The queuing discipline for the model is FIFO. Therefore when the number of packets in the system queue exceeds the queue threshold, packets are dropped.

The arrival process consists of N MMBP-2 source (where $N > 0$) that represents different traffic flows from either the same or different traffic classes. All MMBP-2 sources in the arrival process are superposed together in order to generate an aggregated traffic flow. The number of packets generated at each time slot ranges from 0 to N with each MMBP-2 source generates at maximum one packet per time slot. Each of the MMBP-2 sources can be configured separately.

Fig. 4 shows the state transition diagram of the proposed queuing model. The diagram only captures queue length state transition in single dimension. In fact, the queuing model is a complex model with multi-dimensional transition. Please refers to Figs. 5–8 for an example of multi-dimensional transition breakdown diagram with $N = 2$ for each transition types in Fig. 4. The multi-dimensional diagram is getting complex when N increase.

In order to give a clearer picture of the state transition of the queue length in the system, the queue length state transition matrix ($\tilde{QT}$) are shown in Eq. (6). Please note that the queue size K can be any size greater than zero. For simplicity of explanation and illustration of state transition matrix, $K = 7$ is used. The state transition matrix can be divided into 3 cases that are decided by the number of MMBP-2 sources (N) being superposed and also the queue size (K). The notations of the queue length state transition is explained in Table 1.

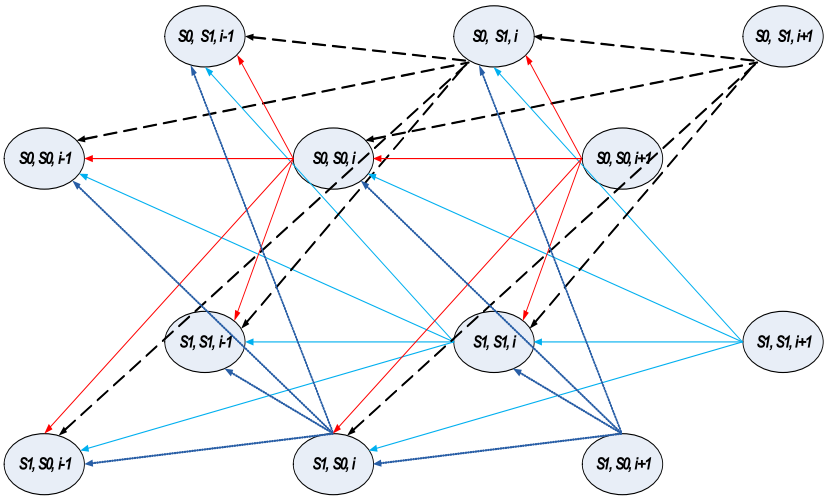


Fig. 6. D elements in $\tilde{QT}$.

$$\tilde{QT} = \begin{cases} \begin{bmatrix} R' & Q(1)' & Q(2)' & Q(3)' & Q(4)' & 0 & 0 & 0 \\ D & R & Q(1) & Q(2) & Q(3) & Q(4) & 0 & 0 \\ 0 & D & R & Q(1) & Q(2) & Q(3) & Q(4) & 0 \\ 0 & 0 & D & R & Q(1) & Q(2) & Q(3) & Q(4)'' \\ 0 & 0 & 0 & D & R & Q(1) & Q(2) & Q(3)^* \\ 0 & 0 & 0 & 0 & D & R & Q(1) & Q(2)^* \\ 0 & 0 & 0 & 0 & 0 & D & R & Q(1)^* \\ 0 & 0 & 0 & 0 & 0 & 0 & D & R^* \end{bmatrix}_{(K+1,K+1)} & \text{for } N < K \\ \begin{bmatrix} R' & Q(1)' & Q(2)' & Q(3)' & Q(4)' & Q(5)' & Q(6)' & Q(7)' \\ D & R & Q(1) & Q(2) & Q(3) & Q(4) & Q(5) & Q(6)^* \\ 0 & D & R & Q(1) & Q(2) & Q(3) & Q(4) & Q(5)^* \\ 0 & 0 & D & R & Q(1) & Q(2) & Q(3) & Q(4)^* \\ 0 & 0 & 0 & D & R & Q(1) & Q(2) & Q(3)^* \\ 0 & 0 & 0 & 0 & D & R & Q(1) & Q(2)^* \\ 0 & 0 & 0 & 0 & 0 & D & R & Q(1)^* \\ 0 & 0 & 0 & 0 & 0 & 0 & D & R^* \end{bmatrix}_{(K+1,K+1)} & \text{for } N = K \\ \begin{bmatrix} R' & Q(1)' & Q(2)' & Q(3)' & Q(4)' & Q(5)' & Q(6)' & Q(7)'^* \\ D & R & Q(1) & Q(2) & Q(3) & Q(4) & Q(5) & Q(6)^* \\ 0 & D & R & Q(1) & Q(2) & Q(3) & Q(4) & Q(5)^* \\ 0 & 0 & D & R & Q(1) & Q(2) & Q(3) & Q(4)^* \\ 0 & 0 & 0 & D & R & Q(1) & Q(2) & Q(3)^* \\ 0 & 0 & 0 & 0 & D & R & Q(1) & Q(2)^* \\ 0 & 0 & 0 & 0 & 0 & D & R & Q(1)^* \\ 0 & 0 & 0 & 0 & 0 & 0 & D & R^* \end{bmatrix}_{(K+1,K+1)} & \text{for } N > K \end{cases} \tag{6}$$

Each of the state transition elements ($R/D/Q$) in the queue length state transition matrix is a 2^N by 2^N matrix which are represented as $R'/R/R^*$, D, $Q(n)'/Q(n)/Q(n)''/Q(n)^*/Q(n)'^*$ in $\tilde{QT}$ (Eq. (6)). As there are a total of N MMBP-2 sources with each MMBP-2 has two distinct states that can be encoded/represented by a one bit binary value 0 (for state $S0$) and 1 (for state $S1$). There will be 2^N combinations of state for N MMBP-2. Therefore, the complexity of the queuing analysis increases with N increases. R represents the state transition where queue length remain unchanged, D represents the state transition where queue length is decreased by one due to a packet departure, while $Q(n)$ state transition means that queue length is increased by n ($1 \leqslant n \leqslant N$). Figs. 5–8 illustrate the queue length state transition elements with 2 MMBP-2 as arrival sources. The queue length state transition element can be obtained through steps below:

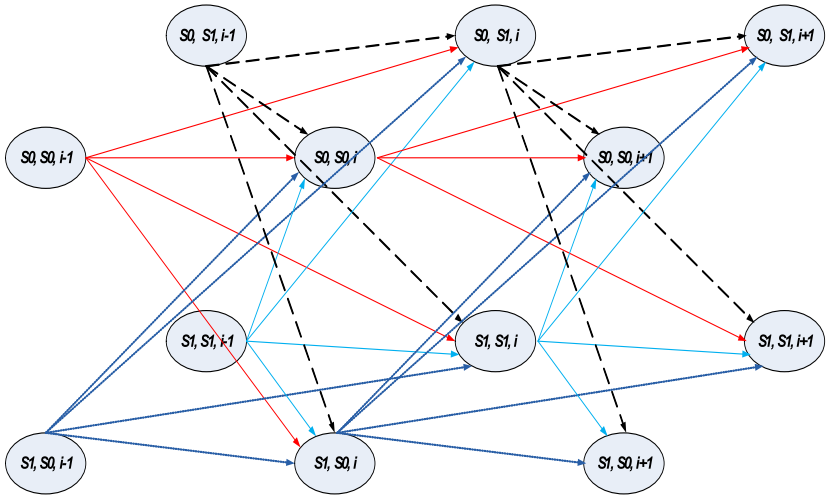


Fig. 7. $Q(1)$ elements in $\tilde{QT}$.

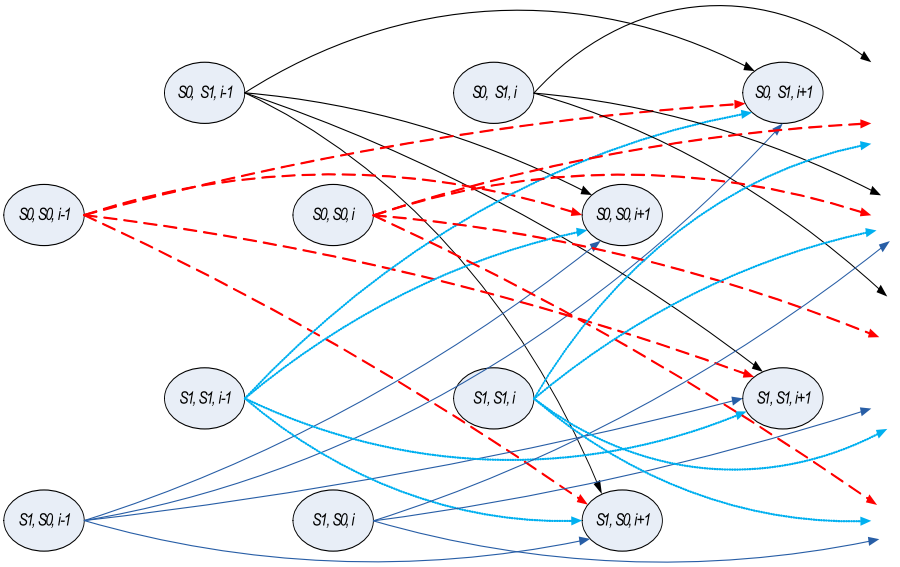


Fig. 8. $Q(2)$ elements in $\tilde{QT}$.

- Derivation of state transition probabilities matrix ($\tilde{S}$) (Eq. (7)) for N MMBP-2 step by step through Eqs. (11)–(12).

$$\tilde{S} = \begin{bmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,2^N} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,2^N} \\ \vdots & \vdots & \cdots & \vdots \\ s_{2^N,1} & s_{2^N,2} & \cdots & s_{2^N,2^N} \end{bmatrix}_{2^N \times 2^N} \tag{7}$$

- Coupling $\tilde{S}$ with arrival or no-arrival probability depending on the state of each MMBP-2 source to derive packet arrivals matrices $\tilde{A}(k)$ of N MMBP-2 (Eq. (8)) through Eqs. (13)–(21). k is the number of packets generated at each time slot. Since that the arrival process consists of N MMBP-2 sources, the number of packets generated at each time slot can range from 0 to N. $\tilde{A}(k)$ is the packet arrival matrices for all possibilities of scenarios.

$$\tilde{A}(k) = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,2^N} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,2^N} \\ \vdots & \vdots & \cdots & \vdots \\ a_{2^N,1} & a_{2^N,2} & \cdots & a_{2^N,2^N} \end{bmatrix}_{2^N \times 2^N} \quad ; \quad \text{for } k = [0..N] \tag{8}$$

Table 1
Notations for queue length transition probability matrix (*Eq.* (6)).

Symbols	Explanation
R'	Special case where queue length will remain unchanged as no packet arrival and no packet departure (empty queue).
R	Queue length will remain unchanged in the following scenarios: 1) No packet departure and no packet arrival at that *TS* 2) 1 packet arrival and 1 packet departure at that *TS*
R^*	Special case where queue length will remain unchanged in the following scenarios: 1) At least $\geqslant 1$ packets arrive into the queue at that *TS* with or without packet departure 2) No packet departure and no packet arrival at that *TS*
D	1 packet departure with no packet arrival at that *TS*.
$Q(n)'$	Special case where queue length will be incremented by n with n packets arrival, no packet departure (empty queue).
$Q(n)'^*$	Special case where no packet departure (empty queue) and queue length will be incremented by n with $> n$ packets arrive into the queue at that *TS*.
$Q(n)$	Queue length is incremented by n in the following scenarios: 1) n packets arrival and no packet departure at that *TS* 2) $n+1$ packets arrival and one packet departure at that *TS*
$Q(n)''$	Queue length is incremented by n where $n = N$ and no packet departure. Note: N is the maximum number of MMBP-2 sources.
$Q(n)^*$	Special case where queue length will be incremented by n even with $\geqslant n$ packets arrive into the queue at that *TS* in the following scenarios, the equation (*current queue length* $+ n = K$) must be satisfied. Note: K is the maximum queue size. 1) n packets arrival and no packet departure 2) With $> n$ packets arrival with or without packet departure

- Derive each of the queue length state transition elements ($QT_{i,j}$) (a.k.a. $R/D/Q(n)$) by coupling the arrivals matrix with departure (β) or no-departure ($1-\beta$) probability (Eqs. (22)–(24)).

The notations shown in Eqs. (9) and (10) are the state transition matrix ($\tilde{P}_i$) and arrival probability matrix ($\tilde{\Lambda}_i$) of each MMBP-2 source represented by i with $i = [1..N]$. The state of ith MMBP-2 is denoted as ST_i. The next state transition probability of ith MMBP-2 is denoted as SP_i; it is determined by its current state and next state as shown in Eq. (11).

$$\tilde{P}_i = \begin{bmatrix} p_i & 1-p_i \\ 1-q_i & q_i \end{bmatrix}, \quad \text{where } i = 1..N \tag{9}$$

$$\tilde{\Lambda}_i = \begin{bmatrix} \alpha_{i1} & 0 \\ 0 & \alpha_{i2} \end{bmatrix}; \quad \text{where } i = 1..N \tag{10}$$

$$SP_i = \begin{cases} p_i & \text{if current } ST_i = S0 \\ & \text{\& current } ST_i = \text{next } ST_i \\ 1-p_i & \text{if curent } ST_i = S0 \\ & \text{\& current } ST_i \neq \text{next } ST_i \\ 1-q_i & \text{if curent } ST_i = S1 \\ & \text{\& current } ST_i \neq \text{next } ST_i \\ q_i & \text{if curent } ST_i = S1 \\ & \text{\& current } ST_i = \text{next } ST_i \end{cases} \tag{11}$$

The derivation of next state transition probability element $S_{m,n}$ are shown in Fig. 9. Indexes of each element in $\tilde{S}$, which are m and n, actually refer to the binary aggregation of current state and next state for all MMBP-2 sources. ith-bit of binary indexes m and n represents the current or next state of ith MMBP-2 source. Both indexes are turned into binary encoding in order to represent the state of each MMBP-2 source. The state transition probability $S_{m,n}$ is the product of transition probability of all MMBP-2 sources as shown in Eq. (12). In Fig. 9, the derivation of $s_{5,22}$ with six MMBP-2 sources is taken as an example for the derivation.

$$s_{m,n} = \prod_{i=1}^{N} SP_i \tag{12}$$

The packet arrivals matrix ($\tilde{A}(k)$) can be derived by associating the arrival probability or no-arrival probability to each MMBP-2 according to its state and also number of arrivals. *SRC* is a set contains of all MMBP-2 sources. Each MMBP-2 source has the same equality in being selected as the source for packet generation. The MMBP-2 combination sets are subsets of *SRC* as shown in Eqs. (15) and (16). The MMBP-2 combinations sets are the list of sources that are randomly selected from the given set of MMBP-2 sources (*SRC*) for packets generation. The number of possible combinations (G) can be determined

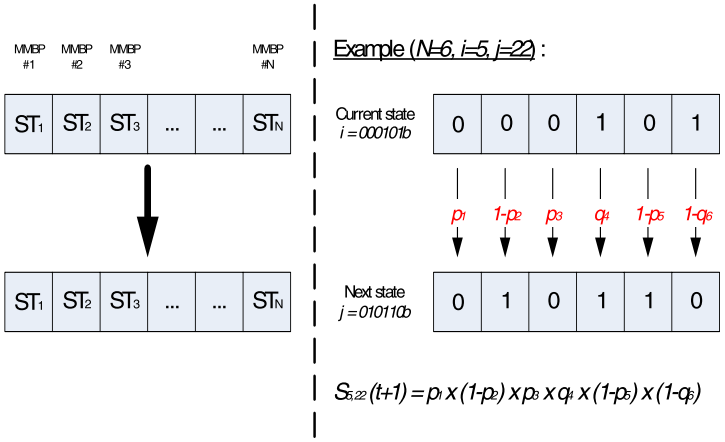


Fig. 9. Derivation of state transition probability elements, $S_{m,n}$ (Eq. (12)).

by Eq. (14). Following this, arrival probabilities of each MMBP-2 sources from Eqs. (15)–(16) are determined to derive $\tilde{A}(k)$ as in Eqs. (17)–(21).

$$SRC = \{1, 2, , 3, \ldots, N\} \tag{13}$$

$$G = C_k^N; \quad \text{for } k = [0..N] \tag{14}$$

$$SRC\_C(k) = \begin{bmatrix} x_{1,1} & \cdots & x_{1,k} \\ \vdots & \ldots & \vdots \\ x_{G,1} & \cdots & x_{G,k} \end{bmatrix}_{G,k}; \quad \text{for } x_{i,j} \in SRC \tag{15}$$

$$\overline{SRC\_C}(k) = \begin{bmatrix} y_{1,1} & \cdots & y_{1,N-k} \\ \vdots & \ldots & \vdots \\ y_{G,1} & \cdots & y_{G,N-k} \end{bmatrix}_{G,N-k}; \quad \text{for } y_{i,j} \in SRC \tag{16}$$

$$AP(k) = \begin{bmatrix} ap_1 \\ \vdots \\ ap_G \end{bmatrix} \tag{17}$$

$$ap_i = \begin{cases} \prod_{j=1}^{k} \alpha_{m1}; & \text{for } m = x_{i,j} \,\&\, ST_m(t) = S0 \\ \prod_{j=1}^{k} \alpha_{m2}; & \text{for } m = x_{i,j} \,\&\, ST_m(t) = S1 \end{cases} \tag{18}$$

$$NAP(k) = \begin{bmatrix} nap_1 \\ \vdots \\ nap_G \end{bmatrix} \tag{19}$$

$$nap_i = \begin{cases} \prod_{j=1}^{k} (1 - \alpha_{m1}); & \text{for } m = y_{i,j} \,\&\, ST_m(t) = S0 \\ \prod_{j=1}^{k} (1 - \alpha_{m2}); & \text{for } m = y_{i,j} \,\&\, ST_m(t) = S1 \end{cases} \tag{20}$$

$$a_{i,j} = \sum_{m=1}^{G} \{s_{i,j} * ap_m * nap_m\} \tag{21}$$

After all the arrivals matrices are obtained, queue length state transition matrix elements $QT_{i,j}$ (from Eq. (6) and Table 1) can be derived by combining the arrivals matrices with departure or no-departure probability (Eqs. (22)–(24)). The derivation also depends on the current queue state QL (current queue length) and the maximum queue size K.

$$R = \begin{cases} \tilde{A}(0) \equiv R' & \text{if } QL = 0 \\ \tilde{A}(0) * (1 - \beta) + \tilde{A}(1) * \beta & \text{if } 0 < QL < K \\ \tilde{A}(0) * (1 - \beta) + \sum_{i=1}^{N} A(i) \equiv R^* & \text{if } QL = K \end{cases} \tag{22}$$

$$D = \tilde{A}(0) * \beta \tag{23}$$

Table 2
Configuration for DTH queuing analysis and simulation.

Scenarios	Experiment configurations
Scenario 1	MMBPs with same configuration, $N = [1..5]$ $\alpha_1 = \frac{0.4}{N}$; $\alpha_2 = \frac{0.5}{N}$; $p = 0.9999$; $q = 0.9999$ Departure probability, $\beta = 0.5$ Required delay, $TD = 7$
Scenario 2	MMBPs with different configuration, $N = 3$ MMBP #1: $\alpha_1 = 0.1$; $\alpha_2 = 0.25$; $p = 0.9999$; $q = 0.9999$ MMBP #2: $\alpha_1 = \alpha_2 = 0.2$; $p = 0.9$; $q = 0.9$ MMBP #3: $\alpha = 0.15$; $\alpha_2 = 0$; $p = 0.5$; $q = 0.5$ Departure probability, $\beta = 0.5$ Required delay, $TD = [5..9]$ with stepping 1
Scenario 3	MMBPs with different configuration, $N = 3$ MMBP #1: $\alpha_1 = 0.30$; $\alpha_2 = 0.45$; $p = 0.9999$; $q = 0.9999$ MMBP #2: $\alpha_1 = \alpha_2 = [0.1..0.3]$ with stepping 0.05; $p = 0.9$; $q = 0.9$ MMBP #3: $\alpha = 0.15$; $\alpha_2 = 0$; $p = 0.5$; $q = 0.5$ Departure probability, $\beta = [0.6..0.8]$ with stepping 0.05 Required delay, $TD = 7$

$$Q(n) = \begin{cases} \tilde{A}(n) \equiv Q(n)' & \text{if } QL = 0 \\ & \text{for all } n \text{ in case } N \leqslant K; \text{ and} \\ & \text{for } n < K \text{ in case } N > K \\ \sum_{i=n}^{N} \tilde{A}(i) \equiv Q(n)'* & \text{if } QL = 0 \\ & \text{for } n = K \text{ in case } N > K \\ \tilde{A}(n) * (1-\beta) + \tilde{A}(n+1) * \beta & \text{if } QL > 0 \,\&\, QL + n < K \\ & \text{for all cases} \\ \tilde{A}(n) * (1-\beta) \equiv Q(n)'' & \text{if } QL > 0 \,\&\, QL + n = K \\ & \text{for } n = N \text{ in case } N < K \\ \tilde{A}(n) * (1-\beta) + \sum_{i=n+1}^{N} A(i) \equiv Q(n)* & \text{if } QL > 0 \,\&\, QL + n = K \\ & \text{for case } n < N \text{ in all cases} \end{cases} \tag{24}$$

The $\tilde{QT}$ matrix (Eq. (6)) derived from Eqs. (7)–(24) will then be used in performing steady state queuing analysis. The steady state queuing analysis is performed by using the Markov Chain Solver algorithms [25,31] to solve the joint steady state probability vector of queue length $\tilde{\pi} = \pi_i$ $(0 \leqslant i \leqslant K)$, which satisfies Eqs. (25) and (26) where $e = [1, 1, \ldots, 1]_K^T$ is the column vector of length K. Solving these two equations yields the steady state vector $\tilde{\pi}$ (Eq. (28)) as derived in Markov Chain Solver algorithm where u is an arbitrary row vector of X and I is an identity matrix with size of $K \times K$. A closed-form function derivation approach was not adopted due to the complexity of the queuing model and also a huge number of parameters involved. Numerical analysis in Matlab is used instead.

$$\tilde{\pi}.\tilde{QT} = 0 \tag{25}$$

$$\tilde{\pi}.e = 1 \tag{26}$$

$$X = I + \tilde{QT}/\min\{QT_{i,j}\} \tag{27}$$

$$\tilde{\pi} = u(I - X + eu)^{-1} \tag{28}$$

Common performance metrics used in the discrete-time queuing model, such metrics are system utilization (ρ), average throughput ($\bar{S}$), average queue length ($\bar{L}$) and average queue delay ($\bar{D}$), can then be derived easily with the resolution of queue length steady state probability vector [24,25,29].

5. Results

Simulation of DTH scheme with multiple MMBP-2 sources has been carried out using Matlab simulation. In order to validate and test the feasibility of the proposed scheme, the simulation has been carried out based on the scenarios listed in Table 2 with the following configurations: $K = 60$, $TW = 200$. Note that each $TW = 10{,}000TS$. Three simulation scenarios are discussed in the following subsections.

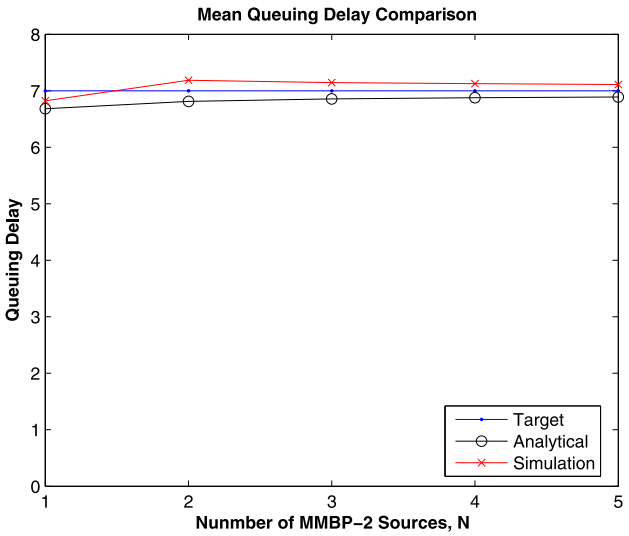


Fig. 10. DTH validation (Scenario 1).

Table 3
SCV and MSE for DTH (Scenario 1).

N	1	2	3	4	5
SCV	0.1379	0.1619	0.3065	0.3482	0.3144
MSE	0.1697	0.1963	0.3274	0.3639	0.3267

5.1. Scenario 1: Different number of sources

Theoretical results are validated via simulation results as shown in Fig. 10. It presents the mean queuing delay experienced in DTH versus the target delay with aggregated traffic flows from same traffic class and with increasing number of traffic flows. The simulation results match the analytical result with small Squared Coefficient of Variation (SCV) and Mean Squared Error (MSE) (Table 3). It shows that DTH is able to maintain the average queuing delay to the target delay with different number of sources. It can be seen that the queuing threshold is adjusted dynamically from time to time in Fig. 11. The fluctuation is due to the adaptive nature of queue threshold.

From the simulation results of Scenario 1, it can be seen that the system could achieve the expected throughput for different number of traffic sources (Fig. 11). However, the packet loss ratio also increases as the number of traffic sources increases as shown in Fig. 12. This can be explained by the burstiness of the arrival process in term of number of packets generated at each time slot. With the increasing number of traffic sources, more packets generated at each time slot and also more packets being dropped due to queue full.

5.2. Scenario 2: Different required average queuing delay

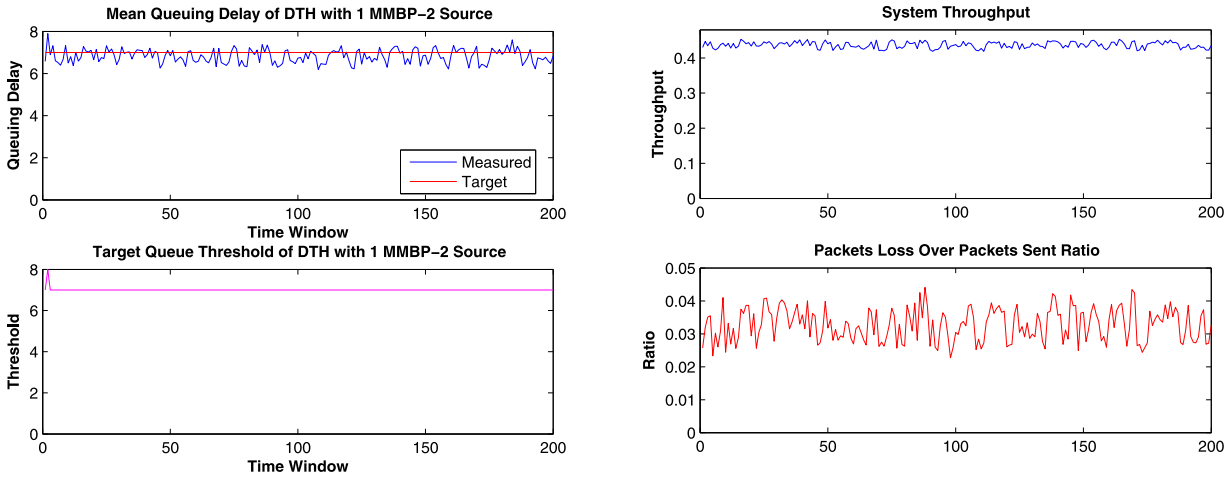
Apart from supporting multiple traffic sources, DTH is capable of maintaining the average queuing delay to different *TD* too as shown in Fig. 13. This is further proven with small SCV and MSE values (Table 4) from the simulation.

For Scenario 2, Figs. 14 and 15 show that packet loss ratio decreases when the required delay value increases. This is due to the upper queuing threshold is relaxed along with the increasing of required delay value. It means more rooms in the buffer queue for packets enqueuing. Consequently, packet loss ratio decreases also.

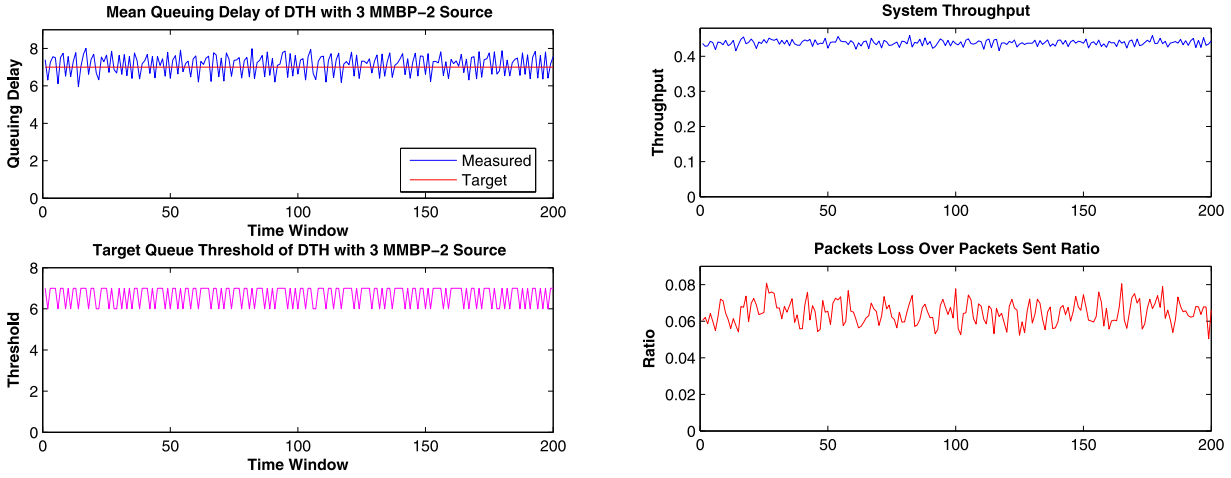
5.3. Scenario 3: Different service rate

Apart from that, DTH is capable of maintaining the average queuing delay to different *TD*. Scenario 3 (Fig. 16) shows the feasibility of DTH scheme to maintain the average queuing delay by taking into account of service rate. These are further proven with small SCV and MSE values (Table 5) from the simulation.

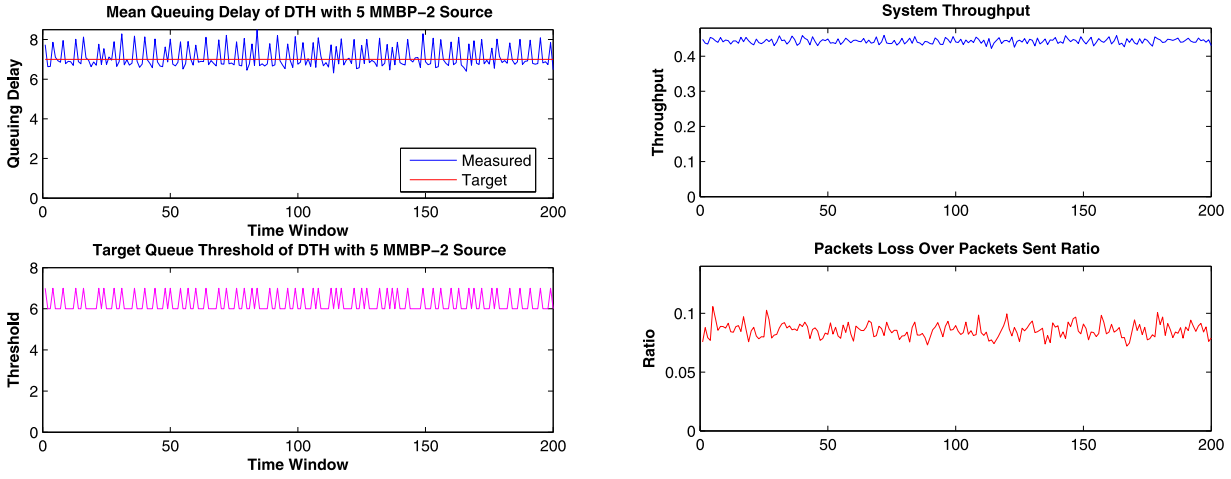
The same for Scenario 3, Figs. 17 and 18 also show that packet loss ratio has a trend of decreasing when the system service rate increases. With the faster rate of packets processing in the system, the queue is drained up faster and leaves more rooms in the buffer queue for new incoming packets. Therefore, packet loss ratio decreases as a consequence of this.



(a) N = 1



(b) N = 3



(c) N = 5

Fig. 11. System performance of DTH (Scenario 1).

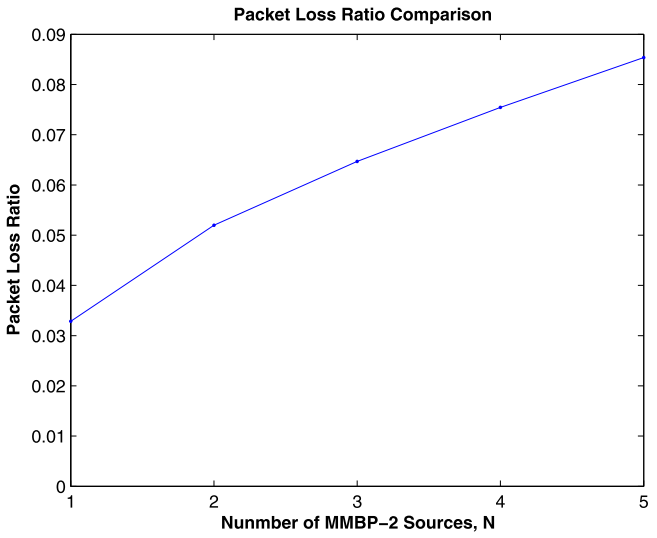


Fig. 12. Packet loss ratio comparison (Scenario 1).

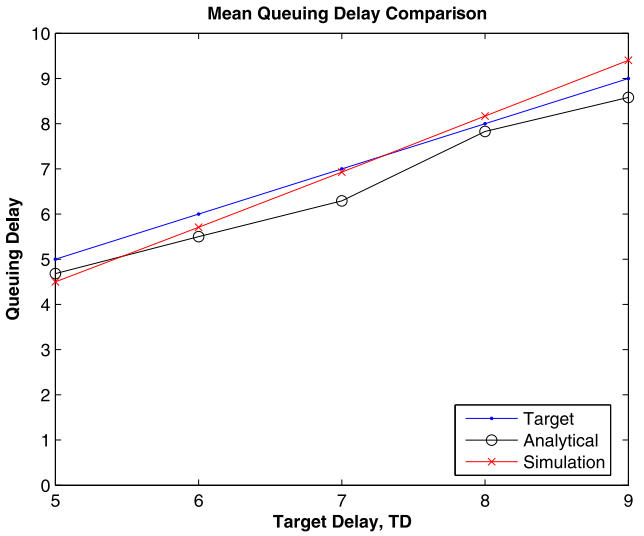


Fig. 13. DTH validation (Scenario 2).

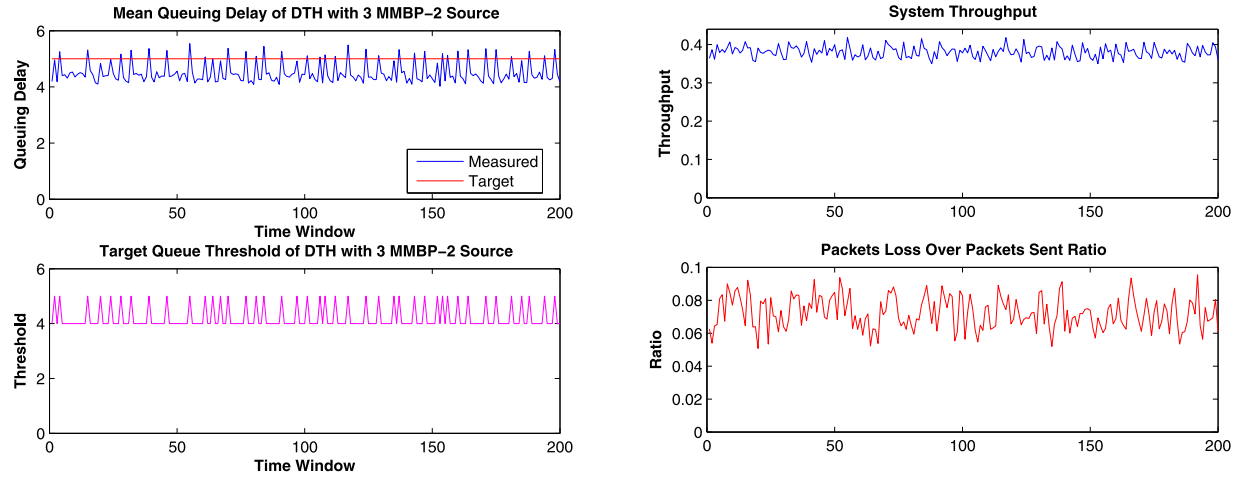
Table 4
SCV and MSE for DTH (Scenario 2).

TD	5	6	7	8	9
SCV	0.3907	0.3456	0.2585	0.2638	0.4475
MSE	0.6395	0.4318	0.2636	0.2914	0.6097

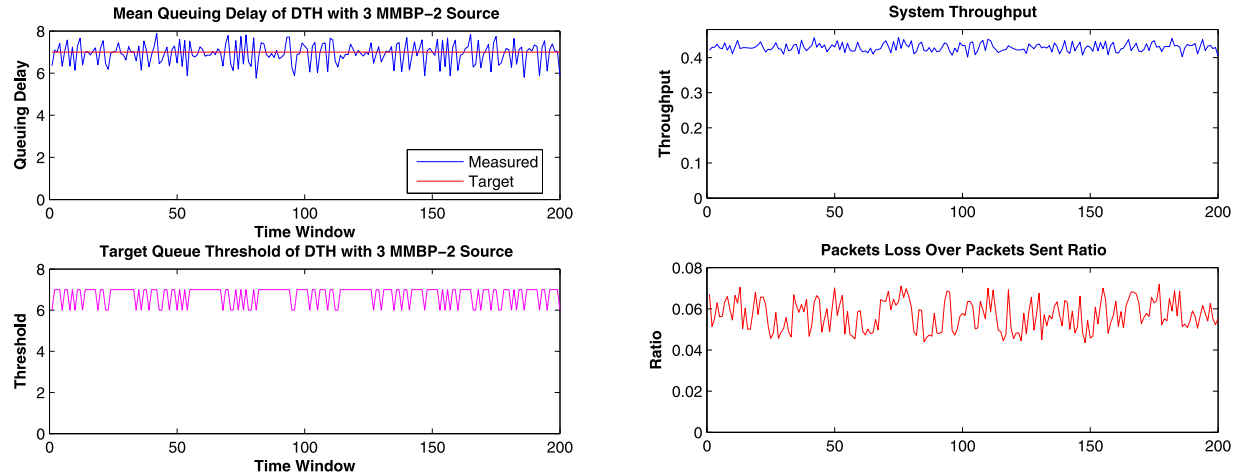
6. Conclusions and future work

This paper describes an adaptive queue management scheme that constraints the average queuing delay in a router to a required value through the mapping relationship between queuing delays and queuing thresholds. The mapping relationship is obtained from an offline discrete-time queuing analysis based on superposition of N MMBP sources that model aggregated Internet traffic. Packets are dropped when queue length exceeds the queue threshold. Consequently, packet loss events can serve as implicit congestion indicators for the sources.

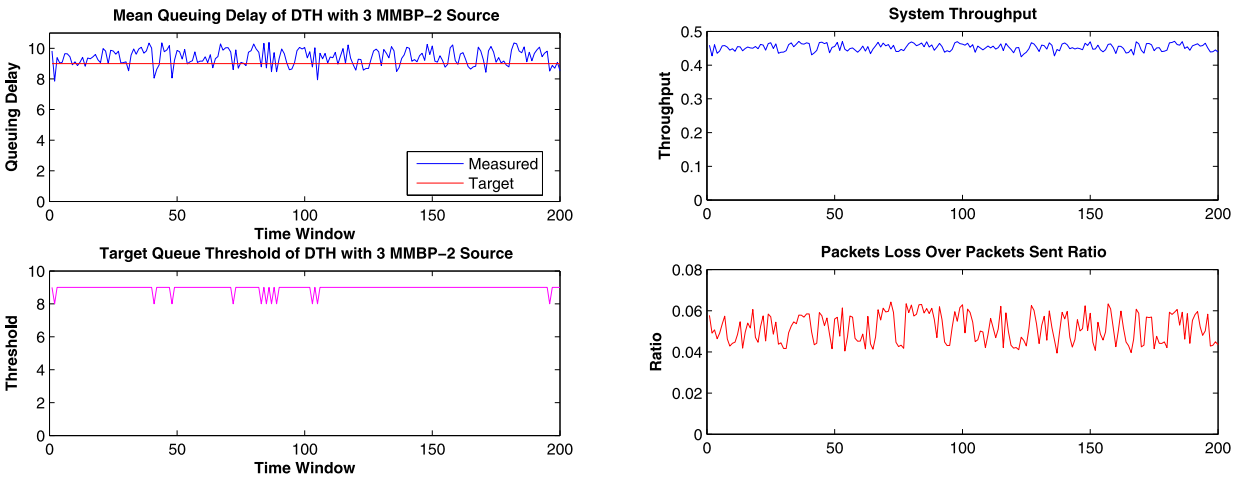
The simulation results show that the proposed DTH solution is able to bound the average queuing delay in a router via a movable queuing threshold using a closed-loop feedback control mechanism. The simulation scenarios also show that the



(a) $TD = 5$



(b) $TD = 7$



(c) $TD = 9$

Fig. 14. System performance of DTH (Scenario 2).

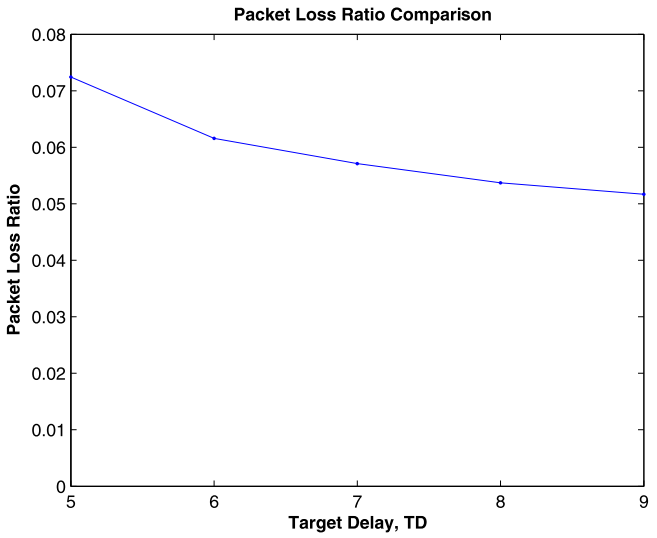


Fig. 15. Packet loss ratio comparison (Scenario 2).

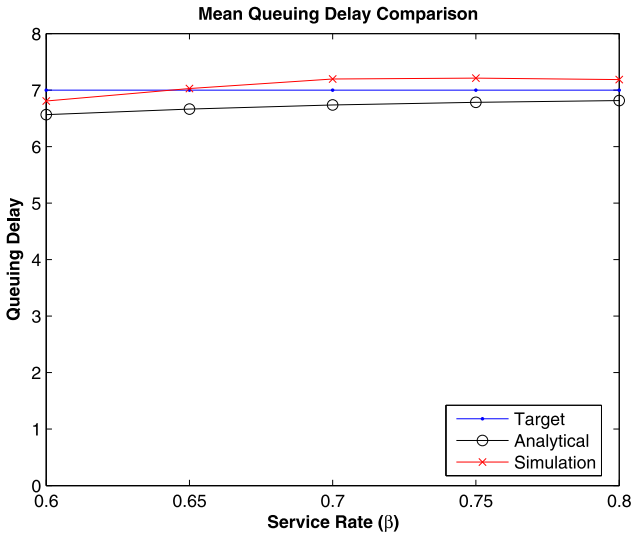


Fig. 16. DTH validation (Scenario 3).

Table 5
SCV and MSE for DTH (Scenario 3).

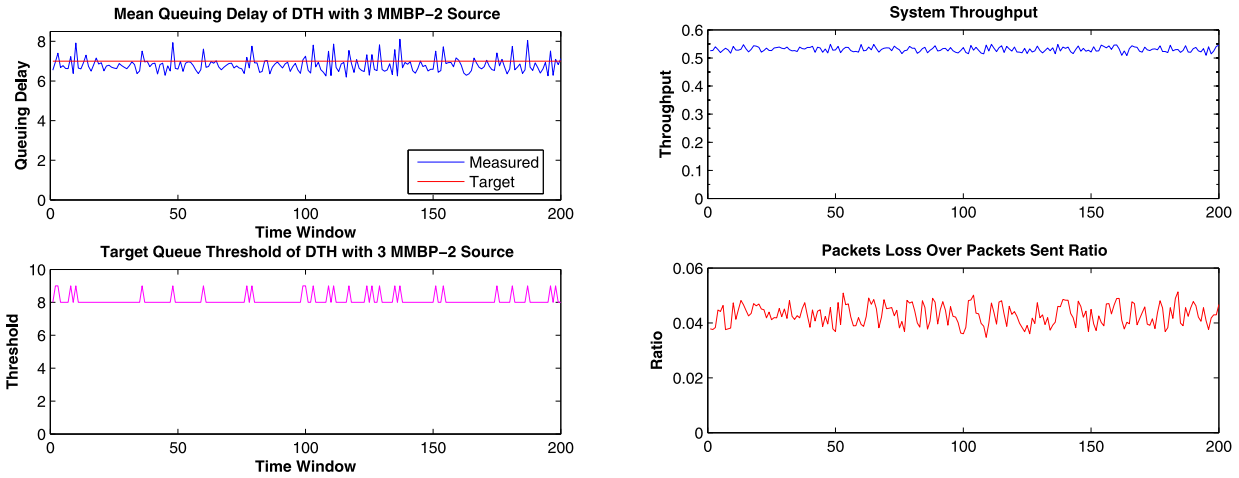
β	0.60	0.65	0.70	0.75	0.80
SCV	0.1784	0.0508	0.1112	0.1959	0.1919
MSE	0.2154	0.0516	0.1497	0.2408	0.2261

simulation results match with analytical results. Therefore, the proposed scheme is feasible in constraining queuing delay at Internet routers.

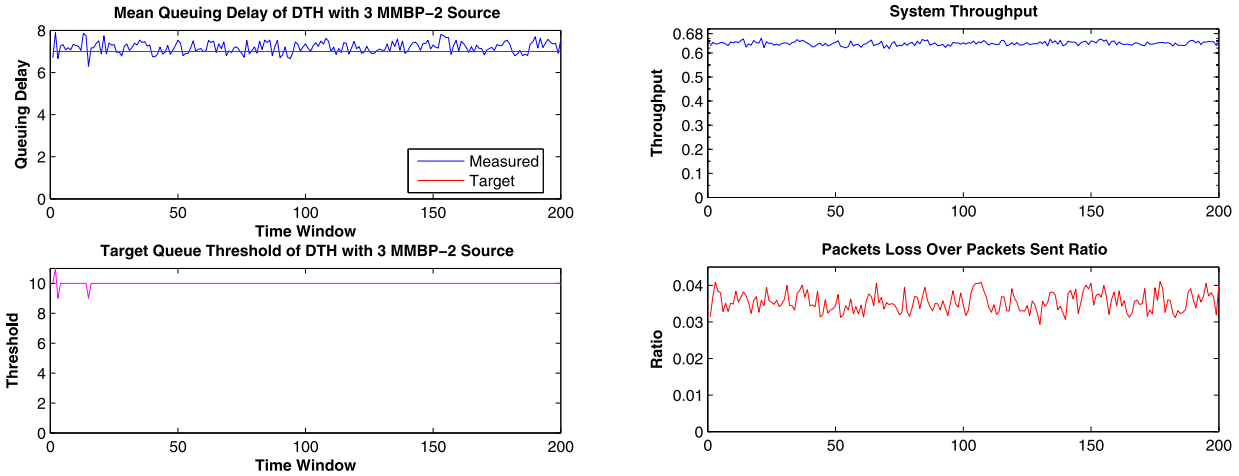
For future work, the DTH mechanism will be implemented on a programmable network processor platform. Performance analysis will then be carried out based on a real-time test-bed.

Acknowledgments

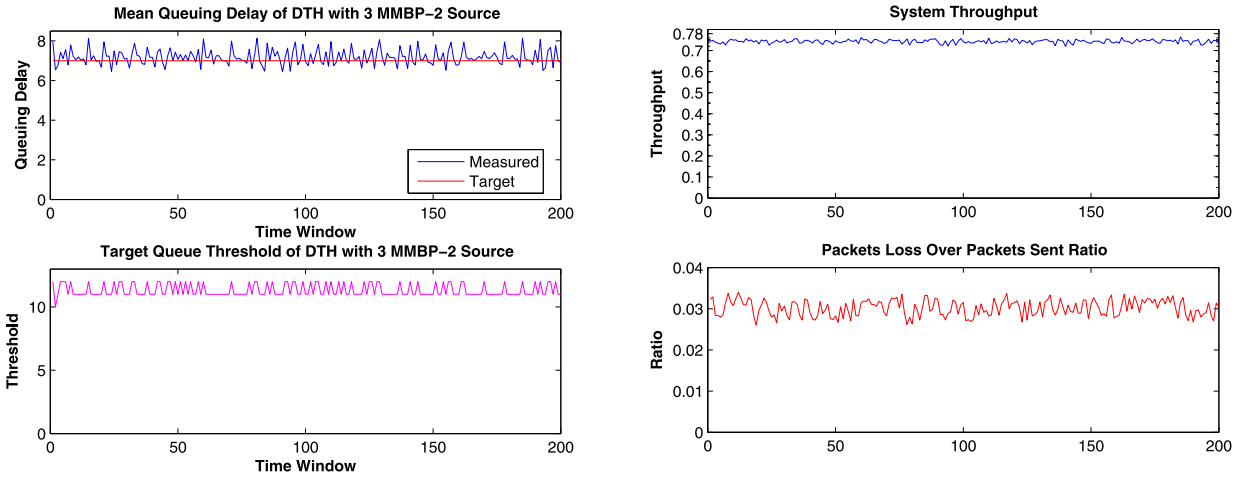
This work is sponsored by the Engineering and Physical Science Research Council (EPSRC), UK, under grant CASE/CNA/07/82 and The Royal Society, UK.



(a) Service Probability = 0.6



(b) Service Probability = 0.7



(c) Service Probability = 0.8

Fig. 17. System performance of DTH (Scenario 3).

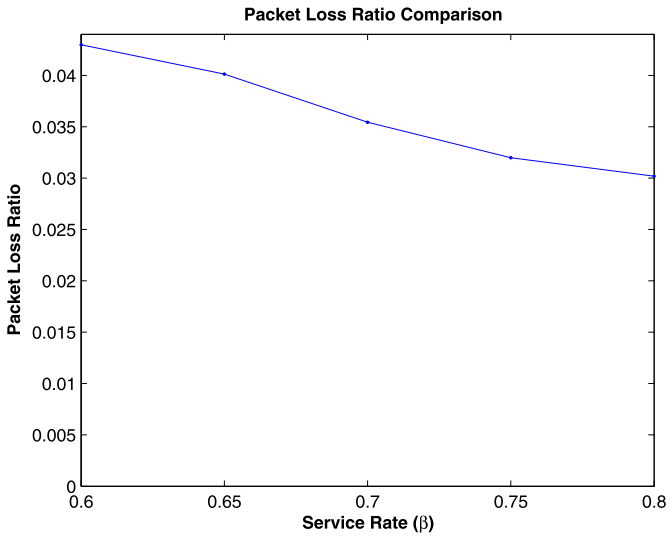


Fig. 18. Packet loss ratio comparison (Scenario 3).

References

[1] H. El-Sayed, A. Mellouk, L. George, S. Zeadally, Quality of service models for heterogeneous networks: overview and challenges, Annals of Telecommunications 63 (11) (2008) 639–668.
[2] D. Malone, P. Clifford, D. Leith, On buffer sizing for voice in 802.11 WLANs, IEEE Communications Letters 10 (10) (Oct. 2006) 701–703.
[3] S. Floyd, V. Jacobson, Random early detection gateways for congestion avoidance, IEEE/ACM Transactions on Networking 1 (4) (Aug. 1993) 397–413.
[4] F. Anjum, L. Tassiulas, Fair bandwidth sharing among adaptive and non-adaptive flows in the Internet, in: INFOCOM '99, Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings, vol. 3, IEEE, Mar. 1999, pp. 1412–1420.
[5] W.-C. Feng, D. Kandlur, D. Saha, K. Shin, A self-configuring RED gateway, in: INFOCOM '99, Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings, vol. 3, IEEE, Mar. 1999, pp. 1320–1328.
[6] S. Floyd, R. Gummadi, S. Shenker et al., Adaptive RED: An algorithm for increasing the robustness of RED's active queue management, August 2001, preprint, available at http://www.icir.org/floyd/papers.html.
[7] B. Zheng, M. Atiquzzaman, DSRED: A new queue management scheme for the next generation Internet, in: IEICE Transactions on Communications, 2006, pp. 764–774.
[8] F. Al-Raddady, M.Woodward, A new adaptive congestion control mechanism for the Internet based on RED, in: 21st International Conference on Advanced Information Networking and Applications Workshops, AINAW '07, vol. 2, May 2007, pp. 934–939.
[9] X. Huang, J. Wu, G. Sun, J. Jing, A new fair active queue management algorithm, in: International Conference on Future Networks, 2009, pp. 191–195.
[10] C. Hollot, V. Misra, D. Towsley, W.-B. Gong, On designing improved controllers for AQM routers supporting TCP flows, in: INFOCOM 2001, 20th Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings, vol. 3, 2001, pp. 1726–1734.
[11] S. Athuraliya, S. Low, V. Li, Q. Yin, REM: active queue management, IEEE Network 15 (3) (May 2001) 48–53.
[12] L. Le, K. Jeffay, F. Smith, A loss and queuing-delay controller for router buffer management, in: 26th IEEE International Conference on Distributed Computing Systems, ICDCS 2006, 2006, pp. 4–4.
[13] R. Pan, B. Prabhakar, K. Psounis, CHOKe - a stateless active queue management scheme for approximating fair bandwidth allocation, in: INFOCOM 2000, Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings, vol. 2, IEEE, 2000, pp. 942–951.
[14] W. Feng, K. Shin, D. Kandlur, D. Saha, The BLUE active queue management algorithms, IEEE/ACM Transactions on Networking (TON) 10 (4) (2002) 513–528.
[15] W. Feng, D. Kandlur, D. Saha, K. Shin, Stochastic fair blue: a queue management algorithm for enforcing fairness, in: INFOCOM 2001, Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings, vol. 3, IEEE, 2001, pp. 1520–1529.
[16] A. Kapadia, W. Feng, R. Campbell, GREEN: A TCP equation-based approach to active queue management, Technical Report UIUCDCS-R-2004-2408, University of Illinois, 2004.
[17] J. Hong, C. Joo, S. Bahk, Active queue management algorithm considering queue and load states, in: 13th International Conference on Computer Communications and Networks, ICCCN 2004, Proceedings, Oct. 2004, pp. 140–145.
[18] T. Karagiannis, M. Molle, M. Faloutsos, Long-range dependence ten years of Internet traffic modeling, Internet Computing, IEEE 8 (5) (Sept.–Oct. 2004) 57–64.
[19] C. Di Cairano-Gilfedder, R. Clegg, A decade of Internet research – advances in models and practices, J. BT Technology 23 (4) (2005) 115–128.
[20] C. Ng, L. Yuan, W. Fu, L. Zhang, Methodology for traffic modeling using two-state Markov-modulated Bernoulli process, Computer Communications 22 (13) (1999) 1266–1273.
[21] C. Ng, B.-H. Soong, Queueing Modelling Fundamentals: With Applications in Communication Networks, second ed., Wiley Publishing, 2008.
[22] A. Andersen, B. Nielsen, A Markovian approach for modeling packet traffic with long-range dependence, IEEE Journal on Selected Areas in Communications 16 (5) (Jun. 1998) 719–732.
[23] L. Muscariello, M. Mellia, M. Meo, M.A. Marsan, R.L. Cigno, Markov models of Internet traffic and a new hierarchical MMPP model, Computer Communications 28 (16) (2005) 1835–1851.
[24] M. Woodward, Communication and Computer Networks, IEEE Computer Society Press, Los Alamitos, CA, 1994.
[25] L. Guan, M. Woodward, I. Awan, Control of queueing delay in a buffer with time-varying arrival rate, J. Comput. System Sci. 72 (7) (2006) 1238–1248.
[26] O. Al-Jaber, L. Guan, X. Wang, I. Awan, A. Grigg, X. Chi, Delay restraining of combined multiple input cross core router, in: 22nd International Conference on Advanced Information Networking and Applications, AINA 2008, March 2008, pp. 355–362.

[27] J. Wang, L. Guan, X. Wang, A. Grigg, I. Awan, I. Phillips, X. Chi, QoS enhancements and performance analysis for delay sensitive applications, in: International Conference on Advanced Information Networking and Applications, AINA '09, May 2009, pp. 331–338.
[28] L. Lim, L. Guan, A. Grigg, I. Phillips, X. Wang, I. Awan, Bounding queuing delay in a router based on superposition of N MMBP arrival process, in: Proceedings of 18th International Conference on Computer Communications and Networks, ICCCN 2009, Aug. 2009, pp. 1–6.
[29] L. Guan, I. Awan, M. Woodward, Stochastic modelling of random early detection based congestion control mechanism for bursty and correlated traffic, Software, IEE Proceedings 151 (5) (2004) 240–247.
[30] X. Yuan, M. Ilyas, Modeling of traffic sources in ATM networks, in: SoutheastCon, 2002, Proceedings IEEE, 2002, pp. 82–87.
[31] W. Fischer, K. Meier-Hellstern, The Markov-modulated Poisson process (MMPP) cookbook, Performance Evaluation 18 (2) (1993) 149–171.