논문 2005-42TC-12-3

멀티미디어 트래픽을 위한 MCDT (Multiple-Class Dynamic Threshold) 알고리즘

(Multiple-Class Dynamic Threshold algorithm for Multimedia Traffic)

김 상 윤*, 이 성 창**, 함 진 호***

(SangYun Kim, Sungchang Lee, and Jinho Ham)

요 약

현재 사용되고 있는 Internet상의 트래픽은 어플리케이션의 종류에 따라 다양한 수준의 QoS 요구조건, 즉, Delay, Loss 그리고 Throughput성능에 대한 요구조건을 요청하고 있다. TCP protocol을 사용하는 FTP서비스나 E-Mail 등의 서비스는 Delay 나 Loss 성능보다는 Throughput성능에 대한 요구가 높은 편이기 때문에 앞의 두 성능악화의 반대급부로서 Throughput 성능을 보장받곤 한다. 반면에, 대부분 UDP protocol을 사용하는 real-time streaming 어플리케이션, 예를 들면, IP telephony, Video conferencing, 혹은 network games등의 어플리케이션은 여타의 것들에 비하여 Throughput 보다는 delay나 loss에 대한 성능을 상대적으로 높게 요구한다. 하지만 현재의 AQM들은 best-effort 서비스에 초점을 맞추고 있다. 즉, throughput 성능을 위하io delay나 loss성능을 희생하고 있기 때문에 다양한 어플리케이션에 적합한 서비스를 제공하기 힘들다. 따라서 본 논문에서는 각 어플리케이션이 필요로 하는 QoS 성능을 고려하여 어플리케이션들을 세 가지 클래스로 분류한 뒤 적합한 QoS 요구조건을 고려한 새로운 AQM 알고리즘을 제안한 뒤, 시뮬레이션을 통하여 다른 AQM 알고리즘과 비교 분석하고 그에 대한 결론을 도출한다.

Abstract

Traditional Internet applications such as FTP and E-mail are increasingly sharing bandwidth with newer, more demanding applications such as Web browsing, IP telephony, video conference and online games. These new applications require Quality of Service (QoS), in terms of delay, loss and throughput that are different from QoS requirements of traditional applications. Unfortunately, current Active Queue Management (AQM) approaches offer monolithic best-effort service to all Internet applications regardless of the current QoS requirements. This paper proposes and evaluates a new AQM technique, called MCDT that provides dynamic and separated buffer threshold for each Applications, those are FTP and e-mail on TCP traffic, streaming services on tagged UDP traffic, and the other services on untagged UDP traffic. Using a new QoS metric, our simulations demonstrate that MCDT yields higher QoS in terms of the delay variation and a packet loss than RED when there are heavy UDP traffics that include streaming applications and data applications. MCDT fits the current best-effort Internet environment without high complexity.

Keywords : AQM, QoS, DT(Dynamic threshold), TCP

* 학생회원, 한국항공대학교, 정보통신 대학원
(Department of Information and Telecommunication Engineering, Graduate School of Hankuk Aviation University, Korea)
** 평생회원, 한국항공대학교 항공전자정보통신공학부
(School of Electronics, Telecommunication, and Computer Engineering, Hankuk Aviation University, Korea)
*** 정회원, 한국전자통신연구원
(Electronics and Telecommunications Research Institute, Deajeon, Korea)
접수일자: 2005년11월30일, 수정완료일: 2005년12월10일

I. Introduction

The Internet today carries traffic for applications with a wide range of delay, loss and throughput requirements, as depicted in Figure 1. Traditional applications such as FTP and E-mail on TCP protocol that are primarily concerned with throughput and can tolerate high delays due to long router queues in exchange for high throughput. On the other hand, emerging applications such as IP telephony, video conferencing and networked games on Tagged UDP have different requirements in terms of throughput and delay than these traditional applications. In these reasons, also, non-delay sensitive and non-throughput sensitive applications on Untagged UDP need control.

In particular, multimedia applications have more stringent delay and loss constraints than throughput constraints. This leaves delay as their major impediment to acceptable quality. Since Data applications are moderately sensitive to delay as well as throughput and are loss-insensitive by using re-transfer techniques, it falls in the middle in terms of the delay and throughput requirement and falls in the left-middle in terms of the loss and throughput requirement[1][6][8].

Unfortunately, current routers do not provide Quality of Service(QoS) adapted to the mixed traffics. Most Active Queue Management(AQM) techniques are either heavy-weight by requiring significant architectural changes or focus on providing higher

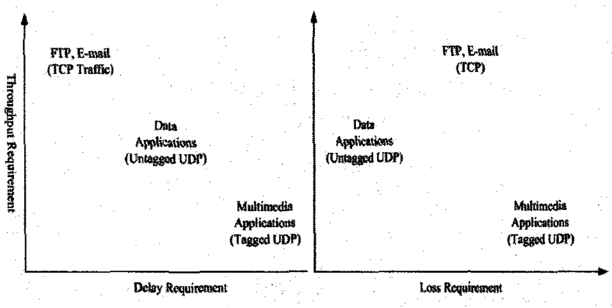


그림 1. Throughput, Delay, Loss에 대한 어플리케이션의 QoS 요구사항

Fig. 1. QoS Requirement of Applications in terms of Throughput, Delay and Loss.

throughput at the router without consideration for queuing delay and loss ratio.

1. Due to the simplicity of the First-In First-Out (FIFO)[4] queuing mechanism, drop-tail queues are the most widely used queuing mechanism in Internet routers today. When faced with persistent congestion, drop-tail routers yield high delay for all application's traffics passing through the bottlenecked router. This best-effort service provides no consideration for multimedia traffics or even data traffic that can be severely affected by high delay. Clearly, drop-tail provides very limited QoS support.

The series of Random Early Detection (RED)[3], like Adaptive RED(ARED)[2] and Flow RED(FRED), the best-known AQM mechanism, attempts to keep the average queue size at the router low while keeping throughput high. By detecting the onset of congestion earlier than drop-tail, RED avoids the global synchronization of TCP traffics that hampers aggregate throughput.

CBT[7] provides class-based treatment with guarantees on bandwidth limits for different classes. However both delay and loss constraints are not considered in CBT, also application delay constraints are not considered.

This paper presents a new AQM technique called Multiple-Class Dynamic Threshold (MCDT) to provide class-based QoS. MCDT will satisfy QoS constraints of Table 1 without adding much complexity and additional policing mechanisms. This study evaluates MCDT via simulation under mixed-traffic scenarios.

The remaining parts of the paper are organized as

표 1. 새로운 AQM 방식을 위한 QoS 요구사항

Table 1. The QoS constraints Table for new AQM technique, two bold 'high' constraints are important QoS elements.

Constraints	TCP traffic (FTP, e-mail)	Untagged UDP (The others)	Tagged UDP (Streaming Applications)
Throughput	High	Medium	Medium
Loss ratio	Medium	Low	High
Delay	Medium	Low	High

follows: The MCDT is explained in section Ⅱ. In section Ⅲ, simulation results are shown. Finally, we make conclusion in section Ⅳ.

Ⅱ. MCDT Algorithm

In this chapter, first, the operation of the dynamic threshold (DT) scheme that provides separate buffer space to TCP, Tagged UDP, and Untagged UDP traffic is described. On the basis of the DT scheme, the proposed AQM algorithm, MCDT (Multiple-Class Dynamic Threshold) is also described

1. Dynamic Threshold

In this study, each traffic class(TCP, Tagged UDP, and Untagged UDP traffics) gets QoS by separated logical buffer that is divided by dynamic buffer threshold. By DT(Dynamic Threshold)[9] method, the relationship of threshold and total shared buffer for each class or flow i is given as

$$T(t)= \alpha\cdot\left(B-Q(t)\right)=\alpha\cdot\left(B-\sum_i Q^i(t)\right) \tag{1}$$

If there are S very active classes, then the total buffer occupancy Q in steady state will be

$$Q= S\cdot T+\Omega \tag{2}$$

The steady-state length of each controlled class threshold can be founded by substituting (2) into (1) and solving for T.

$$T=\frac{\alpha\cdot\left(B-Q(t)\right)}{1+\alpha\cdot S} \tag{3}$$

The amount of shared buffer Θ held in reserve by the algorithm is as follows

$$\Theta=\frac{\left(B-\Omega\right)}{1+\alpha\cdot S} \tag{4}$$

In DT algorithm, every Class has same α value that is selected by user, so every class also has same threshold, T. If we can decide proper α values for each class, then each class can occupy fitting buffer space by dynamic and weighted Threshold.

표 2. DT 알고리즘에서 사용되는 기호 설명
Table 2. Notation of the DT method.

Notation	Expression	Steady-state value
$T(t)$	Threshold	T
$Q(t)$	Sum of $Q^i(t)$	Q
$Q^i(t)$	Queue length of class i	
B	Total buffer space	
S	Number of the very active class	
Ω	Space occupied by the uncontrolled queue $(Q^i<T)$	
Θ	Wastage of DT algorithm	

2. MCDT

If each class has α value respectively, than formula (3) will be

$$T_c=\frac{\alpha_c\cdot\left(B-Q(t)\right)}{1+\sum\alpha_c} \tag{5}$$

Notation T_c and α_c is the threshold and α value of specific class C. In this paper, we will represent α_c as number of arrived packet during a certain period. But, simple count is insufficient to provide QoS as Table 1. TCP class needs high throughput and MMU class needs low loss ratio and low delay. For these requirements, we can set α_c as follows:

$$\alpha_{TCP}=tcp\_count+0.2\times mmu\_count+0.4\times udp\_count \tag{6}$$

$$\alpha_{MMU}=0.8\times mmu\_count \tag{7}$$

$$\alpha_{UDP}=0.6\times udp\_count \tag{8}$$

For the efficient usage of wastage in (4), we add the residue buffer space divided by number of active classes to thresholds.

$$R=\left(B-\sum T_c\right)/S \tag{9}$$

When every update is occurred the weighting factor (we set 0.3) is applied. Finally, MCDT sets each class' threshold as follow

$$T_c = 0.7 \times \left(\frac{\alpha_c \cdot (B - Q(t))}{1 + \sum \alpha_c} + R \right) + 0.3 \times T_{c,Last} \quad (10)$$

As shown in Table 1, traffics are classified by 3 classes; TCP traffic class, Tagged UDP traffic class (MMU traffic class) and Untagged UDP class (UDP traffic class). Using these 3 classes' thresholds, we can divide shared physical buffer into three logical buffers. Briefly, MCDT sets three classes' thresholds

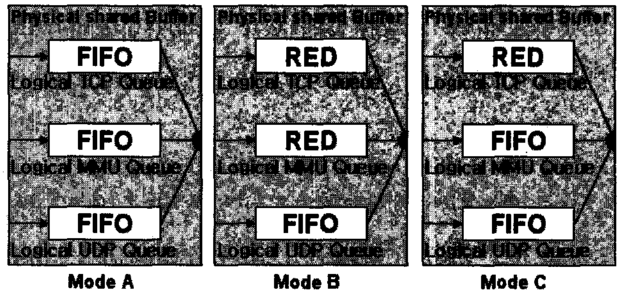


그림 2. MCDT의 세 가지 프로토타입 모드
Fig. 2. 3 proto type modes of MCDT. the difference of each mode is the mechanism of logical queue. By simulation one mode will be chosen.

periodically by counting each class' arriving packets. If a new packet arrives, then MCDT process it as follows:

- Classification: Every packet is classified by its header information like[7].
- Counting: increase the Class' counter (++ count_class).
- Threshold Test: If the length of class' logical queue exceeds the th_class, then drop EOL packet.
- Enque: Enque the packet to class' logical queue.

The MCDT has three prototype modes like Figure 2. The sequence of outgoing packet is the same as sequence of incoming, id. So, there are no additional processes in deque procedure. Mode A is the simplest mode, and mode B, mode C have better QoS performance than mode A with more complexity. In the next section, these three modes are analyzed by simulation.

Ⅲ. Simulations and Results

We ran a simulation for each of RED, CBT, FIFO

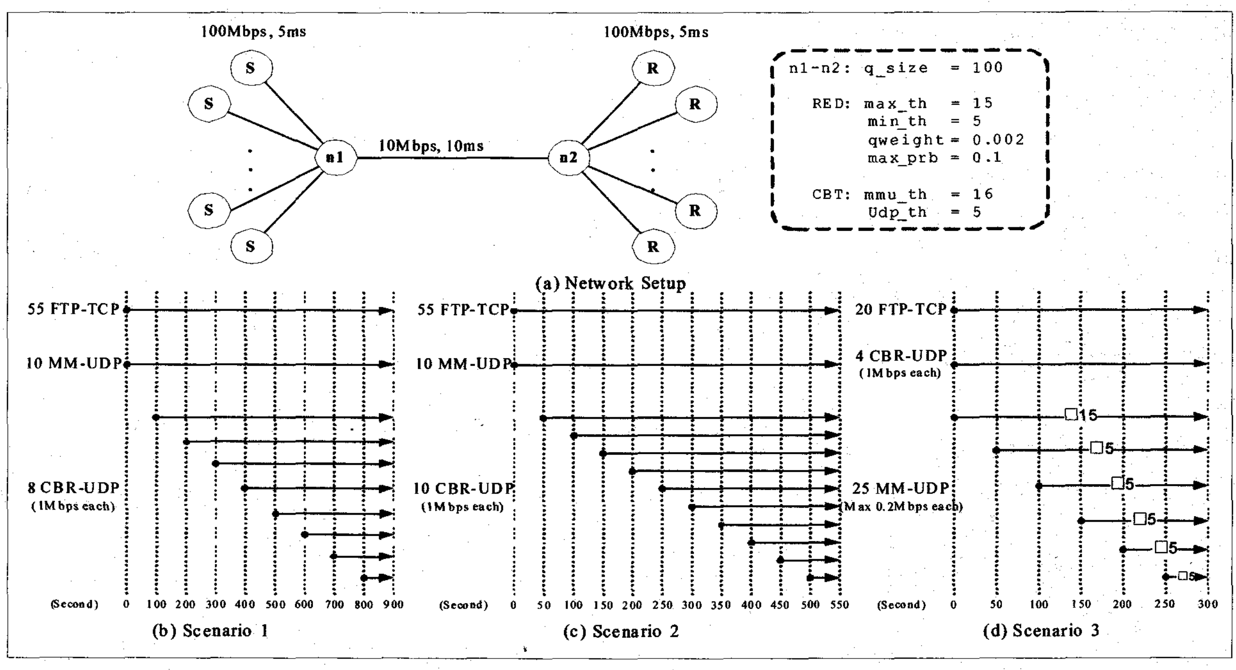


그림 3. 시뮬레이션 시나리오 및 시뮬레이션 네트워크 토폴로지
Fig. 3. The simulation Scenario and Network Setup, each scenario contains the traffic condition of increasing CBR-UDP and MM-UDP respectively.

and MCDT with NS-2[5]. Every simulation had the exactly same settings except the queue management mechanisms of network routers. The network topology and traffic source schedulers are shown in Figure 3. For traffic source, FTP, flow-controlled multimedia traffic generator called MM_APP[7] (tagged) and CBR (untagged) traffic generators were used, where FTP used TCP NewReno and the others used UDP as the underlying transport agent. All the TCP agents were set to have a maximum congestion window size of 20 and all packets in the network were the same size of 1 Kbyte. The MM_APP traffic generators, which react to congestion using 5 discrete media scales with a "cut scale by half at frame loss, up scale by one at RTT" flow control mechanism, used 160, 170, 180, 190 and 200 Kbps for scale 0 to 4 transmission rates. The CBR sources were set to generate packets at a rate of 1 Mbps. Network routers were assigned a 100-packet long physical outbound queue. The RED parameters, which are shown in Figure 3, were chosen from one of the sets that are recommended by Floyd and Jacobson[3]. For CBT, besides the RED parameters, the tagged and untagged class thresholds(denoted as mmu_th and udp_th) were set to 16 packets and 5 packets.

In the Scenario 1, we made a comparison between 3 modes of Figure 2, selected the best mode with considering of QoS performance versus complexity, and recommended it for scenario 2 and 3.

Figure 4 represent drop-rate and Figure 5 is showing just MMU throughput comparison of MCDT modes. The drop-rate and throughput performances of 2 class, FTP-TCP class and CBR-UDP class are

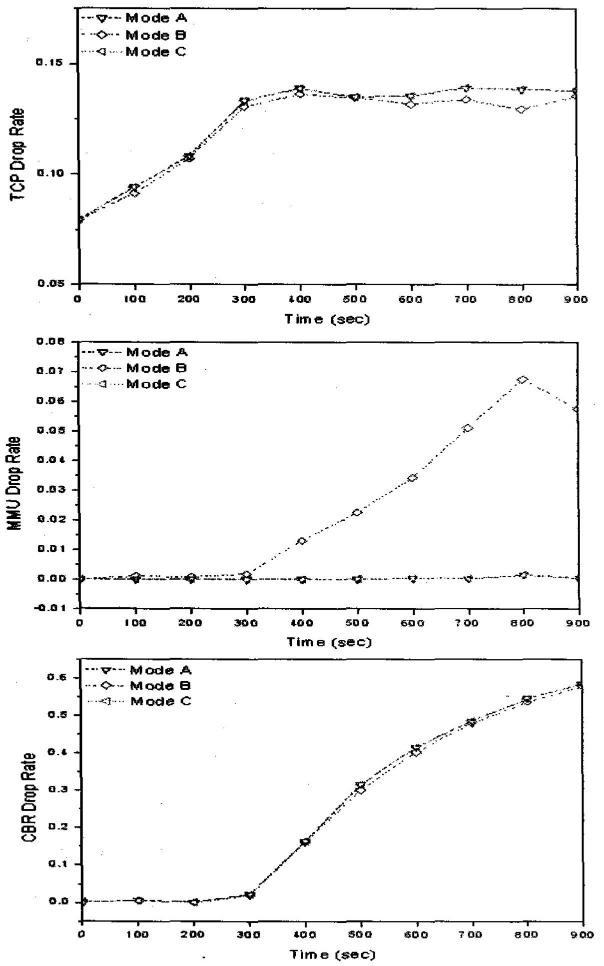


그림 4. 시뮬레이션 시나리오 1에 대한 3가지 모드의 Drop-rate 비교

Fig. 4. The drop-rate comparison of 3 MCDT modes in scenario 1.

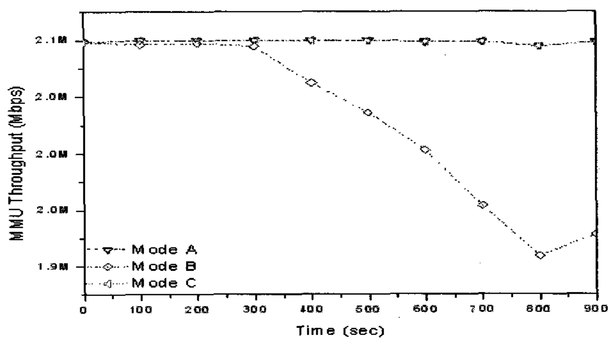


그림 5. 시뮬레이션 시나리오 1에 대한 MMU의 처리량 비교

Fig. 5. The MMU throughput comparison of MCDT modes in scenario 1.

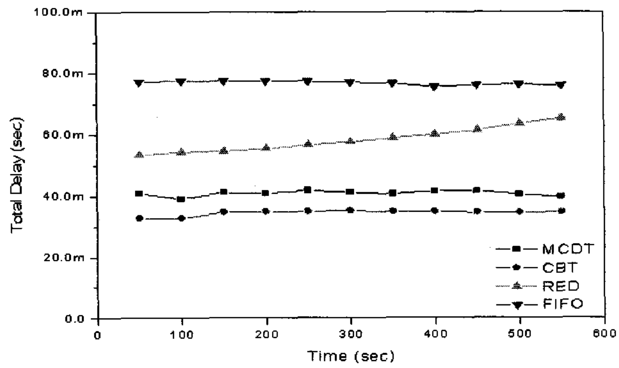


그림 6. 시뮬레이션 시나리오 2에 대한 각 알고리즘의 Delay 비교

Fig. 6. Total Queuing delay(TCP, MMU, UDP) comparison of CBT, RED, FIFO and MCDT in the scenario 2.

similar in all modes.

The mode B offers higher drop-rate and lower throughput of MM-UDP class than other modes. We can analogize reason of the duplicated "cut scale by half" effects. So, we recognize that Mode B is not good choice for MMU. Mode A is most effective mode in a side of complexity. In mode C, we can use MCDT with few additional operations (class determination, counter increase, threshold test) in contrast to RED. So the best mode is Mode A.

The queuing delay comparison of AQM algorithms is presented in Figure 6. The queuing delay of MCDT is less than 40ms. It originates in queue length controlled by dynamic thresholds. The

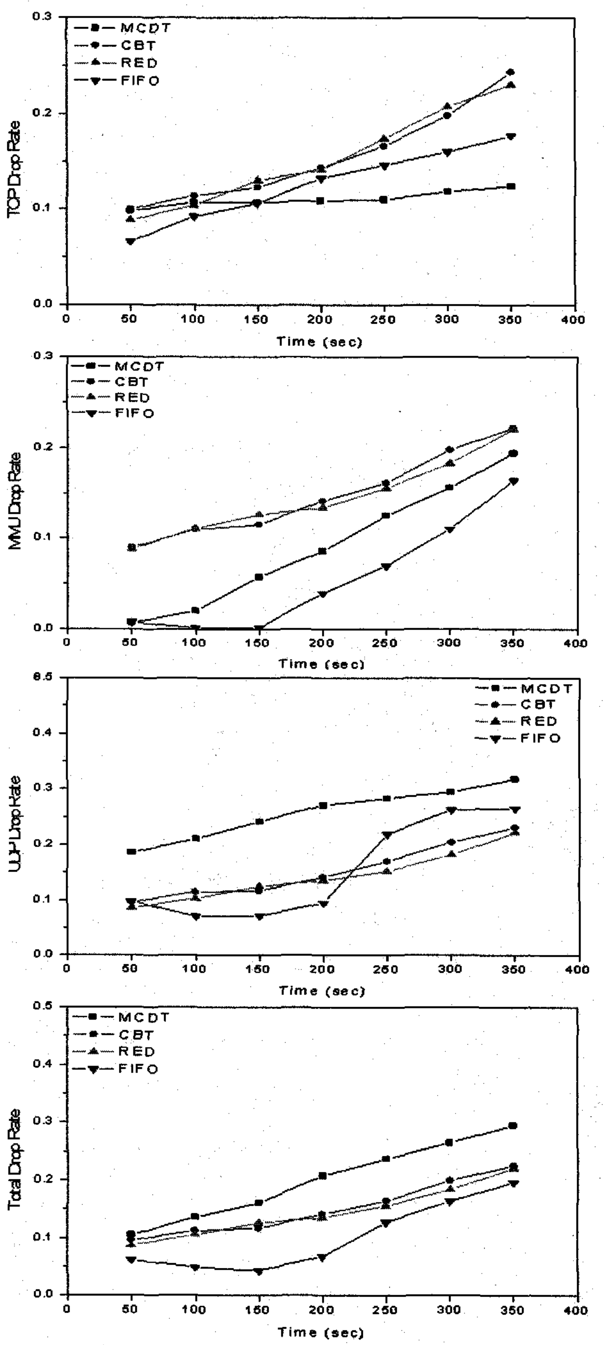


그림 7. 시뮬레이션 시나리오 3에 대한 각 알고리즘의 drop-rate 비교

Fig. 7. The drop-rate comparison of CBT, RED, FIFO and MCDT in scenario 3.

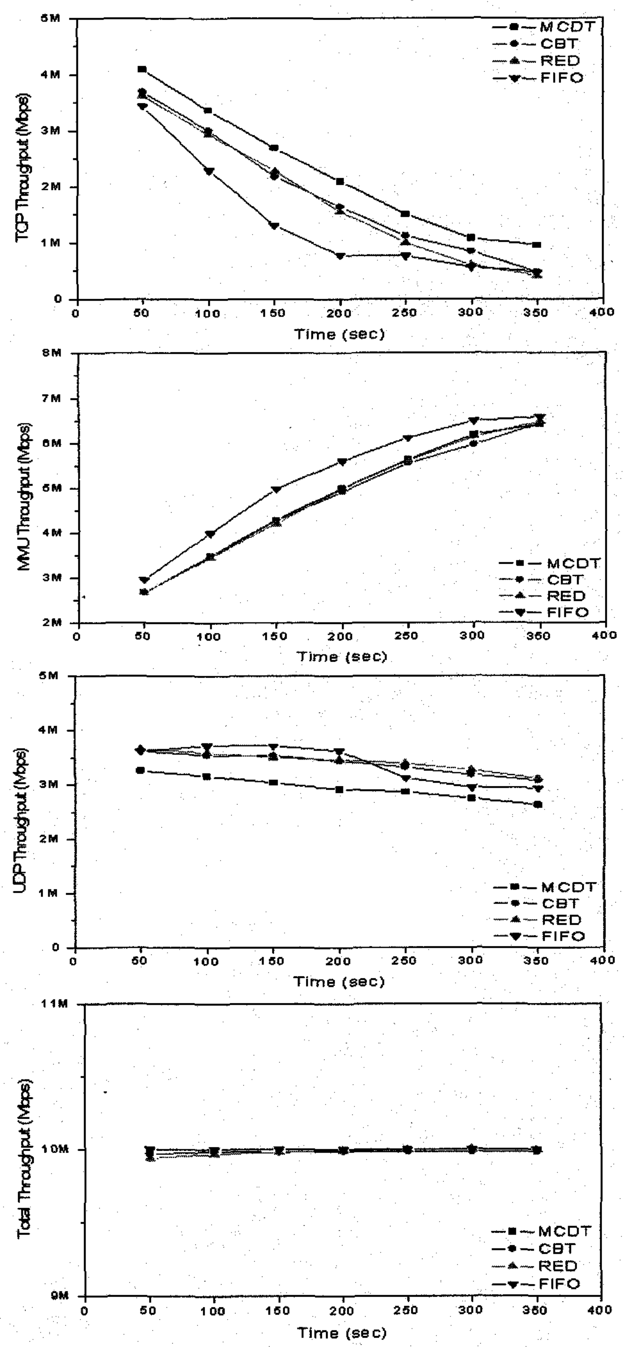


그림 8. 시뮬레이션 시나리오 3에 대한 각 알고리즘의 처리량 비교

Fig. 8. The throughput comparison of CBT, RED, FIFO and MCDT in scenario 3.

MCDT also offers low loss-rate and high throughput of FTP-TCP class. The performance of CBR-UDP is restricted by MCDT, but it is reasonable in our policy.

Figure 7 and Figure 8 represent the performance comparisons of drop-rate and throughput in the scenario 3. Also we could observe the performances provided by MCDT well kept the constraints as Table 1, so total drop rate of proposed algorithm MCDT is more higher than the others. But the throughput performance of FTP-TCP kept high and the drop-rate performance kept low.

V. Conclusion

The traditional Active Queue Management (AQM) algorithms could not satisfy wide constraints of applications. We introduced a new Active Queue Management (AQM) algorithm, called Multiple-Class Dynamic Thresholds (MCDT) as the extension of the Dynamic Threshold (DT) scheme, and investigated the characteristics in comparison with other AQM algorithms. The simulation results show that the MCDT provides higher throughput performance of FTP-TCP, also, the MCDT kept lower delay and drop-rate of MM-UDP than existing AQM algorithms. On the whole, we could conclude that the MCDT had QoS advantages of 2 classes satisfy constraints of each class and these advantages are obtained by fitting dynamic thresholds, especially restricted threshold of CBR-UDP, it is a reasonable result because we focused on QoS constraints of 2 classes, FTP-TCP and MM-UDP. And the MCDT could provide higher QoS than established AQM algorithms with a few additional operations compared to RED and with similar to CBT.

References

[1] J-C. Bolot, S. Fosse-Parisis, and D. Towsley. Adaptive FEC-Based Error Control for Internet Telephony. In Proceeding of IEEE INFOCOM, March 1999.

[2] SallyFloyd, Ramakrishna Gummadi, and Scott Shenker. Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management. Under submission, http://www.icir.org/floyd/papers/adaptiveRed.pdf, 2001.

[3] S.Floyd and V. Jacobson. Random Early Detection Gateway for Congestion Avoidance. IEEE/ACM Transactions on Networking, August 1993.

[4] G. Iannaccone, M. May, and C. Diot. Aggregate Traffic Performance with Active Queue Management and Drop from Tail. ACM Computer Communication Review, July 2001.

[5] University of California Berkeley. The Network Simulator - NS-2. Version 2.26. Internet site is http://www.isi.edu/nsnam/ns/.

[6] C. Padhey, k. Christensen, and W. Moreno. A New Adaptive FEC Loss control Algorithm for Voice Over IP Applications. In Proceedings of IEEE International Performance, Computtingand Communication Conference, February 2000.

[7] Mark Parris, Kevin Jeffay, and F. Smith. Lightweight Active Router-Queue Management for Multimedia Networking. In Proceedings of Multimedia Computing and Networking (MMCN), SPIE Proceedings Series, January 1999.

[8] K. Park and W. Wang. QoS-Sensitive Transport of Real-Time MPEG Video Using Adaptive Forward Error Correction. In Proceedings of IEEE Multimedia Systems, pages 426-432, June 1999.

[9] A. K. Choudhury and E. L. Hanhe, "Dynamic queue length threshold for shared memory packet switches", IEEE/ACM Trans, Networking, vol. 6, pp. 130-140, Apr. 1998.

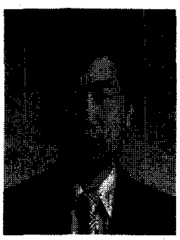
저 자 소 개

김 상 윤(정회원)

2003년 한국항공대학교 정보통신 공학과 학사 졸업.

2005년 한국항공대학교 정보통신 공학과 석사 졸업.

<주관심분야 : 네트워크 프로토콜, RFID, 센서 네트워크>



함 진 호(정회원)

1982년 한양대학교 전자공학과 학사

1984년 한양대학교 전자공학과 석사

1998년 한양대 전자공학과 박사

1984년~현재 한국전자통신연구원 BcN연구단 책임연구원

<주관심분야 : BcN, 멀티미디어 서비스, 통신망 구조, 통신망 프로토콜>

이 성 창(평생회원)

1976년~1983년 경북대학교 전자공학과 학사

1983년~1985년 한국과학기술원 전기및전자 공학과 석사

1985년~1987년 한국과학기술원 시스템공학센터 연구원

1987년~1991년 Texas A&M University 공학 박사

1992년~1993년 한국전자통신연구원

1993년~현재 한국항공대학교 정보통신공학과 교수

<주관심분야 : BcN, 홈 네트워크, 유비쿼터스 네트워크>