

논문 2011-48CI-6-16

# 멀티미디어 데이터 특성을 고려한 동적 큐 관리 알고리즘 연구

## ( A Study on Adaptive Queue Management Algorithm Considering Characteristics of Multimedia Data )

김 지원\*, 윤 정 희\*, 장 은 미\*, 최 상 방\*\*

( Jiwon Kim, Jeonghee Yun, Eunmee Jang, and Sangbang Choi )

### 요 약

멀티미디어 서비스는 많은 대역폭과 실시간 처리가 필요하기 때문에 데이터의 지연과 손실에 민감한 특성을 갖는다. 따라서 네트워크의 혼잡상황으로 인한 데이터의 손실은 사용자에게 QoS를 보장해주지 못한다. 이러한 문제들을 해결하기 위해 네트워크의 혼잡상황과 멀티미디어 데이터의 특성을 고려하여 사용자에게 우수한 품질의 멀티미디어 서비스를 제공할 수 있는 알고리즘의 연구가 필요하다. 본 논문에서는 기지국의 큐 혼잡정도에 따라 패킷을 확률적으로 제거하는 APQM(Adaptive Priority Queue Management) 알고리즘을 제안한다. 기존의 알고리즘과의 비교를 통해 제안한 알고리즘이 네트워크의 혼잡상황을 줄이고, 중요도가 높은 패킷을 많이 수신함으로써 멀티미디어 서비스의 품질이 개선된 것을 확인할 수 있었다.

### Abstract

Multimedia streaming service is susceptible to loss and delay of data as it requires high bandwidth and real time processing. Therefore QoS cannot be guaranteed due to data loss caused by heavy network traffic. To deal with these problems, a study on algorithm that can provide high quality multimedia service by considering both network congestion and characteristics of multimedia data is required. In this paper, we propose APQM algorithm which probabilistically removes packet by the congestion level of the queue in wireless station. The comparison with other existing scheduling algorithms shows tht congestion in the network is reduced and multimedia service quality of the proposed algorithm is improved.

**Keywords :** multimedia data, QoS, queue managemet algorithm

### I. 서 론

지난 5년간 글로벌 인터넷 트래픽은 인터넷 접속이 가능한 단말기와 모바일 인터넷의 확산, 인터넷 사용자의 급증, 비디오 콘텐츠 증가 등으로 다양화되었으며, 향후 지속적인 성장이 이루어질 것으로 전망하고 있다. 특히, 인터넷 트래픽 중에서도 비디오를 중심으로 한 트래픽은 파일 공유(P2P), 웹/데이터보다 큰 비중을 차

지한다<sup>[1]</sup>. 인터넷에서 비디오/오디오 스트리밍 등을 제공하는 실시간 멀티미디어 서비스는 실시간 화상회의, 원격의료등 많은 분야에서 사용되며 손실없는 트래픽 전송과 그에 따른 QoS(Quality of Service) 보장에 대한 연구가 중요시 되고 있다.

현재 네트워크는 사용자 증가와 더불어 증가한 트래픽으로 인하여 혼잡(congestion) 상황이 발생하게 된다. 실시간 멀티미디어 전송은 네트워크 상황에 의해서 서비스 품질이 결정되는데, 이러한 혼잡상황은 큐잉 지연과 패킷 손실을 초래하여 사용자에게 QoS를 보장해주지 못하게 된다. 사용자가 직접 보고 듣는 멀티미디어

\* 학생회원, \*\* 평생회원, 인하대학교 전자공학과 (Dept. of Electronic Engineering Inha University) 접수일자:2011년7월18일, 수정완료일:2011년11월1일

서비스는 제한된 시간내에 서비스되어야 불편함을 느끼지 않는다. 따라서 네트워크의 혼잡상황으로 인한 데이터 손실과 지연은 사용자에게 QoS를 보장해주지 못한다<sup>[2]</sup>. 실시간 멀티미디어 트래픽의 효과적인 전송은 시간적인 특성과 지속적인 대역폭 차지 등의 멀티미디어 트래픽 특성을 만족시키는 전송을 의미한다. 하지만, 현재의 인터넷은 최선의 전송(best-effort)을 목표로 하기 때문에 실시간 멀티미디어 트래픽의 특성을 보장하는 것이 쉽지 않다. 또한 멀티미디어 트래픽은 특성상 UDP를 사용하여 전송하므로, 트래픽의 혼잡과 흐름제어(flow control)는 응용프로그램 자체에 의존하게 된다. 현재 라우터로 유입되는 패킷들은 Drop tail 방식으로 처리된다. 이 방법은 큐에 저장 공간이 있을 때 들어오는 패킷은 모두 저장하고, 큐에 저장 공간이 없을 때 들어오는 패킷은 모두 제거한다. 가장 간단하지만 플로(flow)간의 불공정한 링크 할당, 전역 동기화(global synchronization)에 따른 평균지연 시간의 증가등으로 네트워크의 혼잡이 가중된다. 이러한 문제점을 해결하기 위해 AQM(Active Queue Management)이 발달하였고 RED[3]알고리즘이 제안되었다. 이 알고리즘은 큐의 혼잡상황 발생 시 정해진 확률로 큐에서 패킷을 제거하여 네트워크의 혼잡과 큐 지연을 예방하는 것이다. 이 방법은 효율적인 혼잡제어를 위해 들어오는 모든 패킷을 수용 가능한 충분한 크기의 큐 공간을 확보해야 한다. 하지만 큐 크기가 커지면 큐잉으로 인한 전송시간 지연으로 실시간 트래픽에 대해서 좋은 서비스를 해주기 어렵다. 현재 이 알고리즘을 기반으로 변형 또는 보완한 알고리즘으로는 ARED(Adapted RED)<sup>[4]</sup>, WRED(Weighted RED)<sup>[5]</sup>, ARIO(Adaptive RED with In and Out)<sup>[6]</sup>, BLUE<sup>[7]</sup>, AFD(Approximate Fair Dropping)<sup>[8]</sup>, CHOKe<sup>[9]</sup>, PQM<sup>[10]</sup> 등이 있다.

본 논문에서는 멀티미디어 서비스의 품질 향상을 위한 APQM(Adaptive Priority Queue Management) 알고리즘을 제안한다. ARED 알고리즘과 PQM 알고리즘의 단점을 보완한 것으로, 네트워크의 혼잡정도에 따라 멀티미디어 데이터의 특성을 고려하여 패킷을 제거하는 것이다. 기지국의 큐가 혼잡하면, 혼잡정도를 파악하여 큐의 상태가 이전보다 원활해지면 지연시간이 큰 프레임 패킷을 큐에서 제거한다. 이것은 불필요한 데이터의 전송을 방지하고 패킷수신을 위한 큐 공간을 확보하기 위한 것이다. 반면 기지국의 큐 혼잡이 가중되면, 해당 멀티미디어 플로 내의 패킷 중에서 해당 패킷보다 우선

순위가 낮은 패킷을 제거한다. 이것은 높은 우선순위의 패킷 보호로 멀티미디어 서비스의 품질을 향상시킬 수 있다. 이때 멀티미디어 패킷의 제거는 자신의 전후에 있는 프레임 패킷의 영향을 받는 특성을 고려한다. 큐에 저장된 패킷은 큐의 끝에서부터 제거하되 수신된 패킷보다 낮은 우선순위의 패킷을 제거한다. 제안한 알고리즘의 성능 평가는 NS-2(Network Simulator-2)<sup>[12]</sup>를 이용하였다. 기지국의 큐에 RED, ARED, PQM, 그리고 제안된 알고리즘을 적용하여 수신단에서 수신한 비디오의 PSNR(Peak Signal To Noise Ratio)과 패킷 수신 확률을 비교하였다. 비교결과 APQM 알고리즘이 큐의 혼잡정도를 고려하고 중요도가 높은 패킷을 많이 수신하여 기존 알고리즘보다 성능이 향상되었고, 멀티미디어 서비스 제공에 불필요한 패킷 제거로 큐의 혼잡상황을 완화한 것을 확인할 수 있었다.

본 논문은 다음과 같이 구성되었다. II장에서는 멀티미디어의 특징과 기존의 큐 관리 알고리즘을 소개하고 III장에서는 본 논문에서 제안한 큐 관리 알고리즘을 설명한다. IV장에서는 시뮬레이션을 통해 제안된 큐 관리 알고리즘의 성능 분석을 기술하고, 마지막으로 V장의 결론으로 본 논문을 마무리한다.

## II. 관련 연구

이 장에서는 멀티미디어 데이터에 대해 소개하고 기존에 연구되어온 큐 관리 알고리즘에 대해 설명한다.

멀티미디어 서비스를 위한 인코딩 방식 중 하나인 MPEG은 전송 오류율이 높은 무선 환경에서의 영상 전송에 적합하다. MPEG 인코더는 I(Infra-coded), P(Predicted), 그리고 B(Bidirectional) 프레임으로 구성된다. 그리고 주어진 비디오 순서에 따라 모든 MPEG 프레임을 압축하기 위해 같은 GOP 구조를 반복적으로 사용한다. I 프레임은 가장 중요한 데이터로 손실되면 손실된 MPEG 프레임과 GOP내의 모든 프레임 또한 손실된다. P 프레임이 손실되면 손실된 P 프레임을 포함하여 P 프레임 전의 B 프레임들과 다음에 오는 모든 프레임이 손실된다. 하지만 B 프레임은 다른 MPEG 프레임에 영향을 주지 않는다. 따라서, 어떤 패킷이 손실되었는가에 따라 복구 가능한 프레임에 미치는 영향은 최대 FPS(Frame Per Second)배의 차이가 발생하고, 네트워크의 자원 낭비에 미치는 영향도 큰 차이가 발생한다<sup>[11]</sup>. 그러므로 프레임 패킷에 따라 차등 대우가 필요

하며, 이것으로 네트워크의 부하를 줄이고 더 높은 신뢰성을 보장할 수 있게 된다.

다음은 지금까지 연구된 네트워크에서의 혼잡상황을 관리하는 알고리즘 중 제안한 알고리즘과의 성능 비교에 사용된 알고리즘들에 대해 설명한 것이다.

## 2.1 RED 알고리즘

네트워크의 혼잡상황 발생 시, 큐에 도착하는 패킷을 확률적으로 제거하여 혼잡을 완화시키기 위한 알고리즘이다. 즉,  $Q_{avg}$  (평균 큐 크기)를 탐색하여 혼잡상황이 감지되면, 큐의 평균 크기를 기준으로 계산된 확률로 패킷을 제거한다.  $Q_{avg}$ 는 패킷이 큐에 도착할 때마다 계산하고, 미리 정해진  $th_{min}$  (최소 임계값) 및  $th_{max}$  (최대 임계값)와 비교한다.  $Q_{avg}$ 가  $th_{min}$ 보다 낮은 비혼잡 영역에서는 큐의 사용률이 낮다고 판단하여, 라우터에 들어오는 모든 패킷을 큐에 저장한다. 반면,  $Q_{avg}$ 가 두 임계값 사이에 있으면 패킷을 확률적으로 제거하고,  $th_{max}$  이상이면 입력 패킷을  $Q_{avg}$ 가  $th_{max}$ 보다 낮아질 때까지 큐로 들어오는 패킷을 제거한다.

$Q_{avg}$ 는 식 (1)의 EWMA(Exponential Weighted Moving Average)를 이용하며, 패킷이 도착할 때마다 계산한 값을 네트워크 혼잡제어에 적용한다. 여기서,  $W_q$ 는 현재의 큐 길이가  $Q_{avg}$ 에 미치는 영향을 결정하는 가중치이고,  $Q_{size}$ 는 현재 큐의 길이이다. 식 (2)는 패킷 제거 확률을 계산하는 방법이다.  $P_b$ 는  $th_{min}$ 에서  $th_{max}$ 까지 선형적으로 증가하고,  $P_{max}$ 는 확률  $P_b$ 의 최대값을 정하기 위한 임의의 상수이다. 식 (3)은 확률  $P_b$ 에  $count$ 를 고려한 최종 패킷 제거 확률이다.  $count$ 는 큐에 패킷이 들어올 때마다 증가하고, 패킷이 제거되면 초기화된다. 기지국의 큐에 혼잡상황이 발생하면  $Q_{avg}$ 에 따라 패킷 제거 확률값을 적용하게 된다.

$$Q_{avg} = (1 - W_q)Q_{avg} + W_q \times Q_{size} \quad (1)$$

$$P_b = P_{max} \frac{(Q_{avg} - th_{min})}{(th_{max} - th_{min})} \quad (2)$$

$$P_a = \frac{P_b}{1 - count \cdot P_b} \quad (3)$$

이 알고리즘은 링크의 혼잡도와 설정된 파라미터인  $Q_{avg}$ ,  $P_{max}$ ,  $th_{min}$ , 그리고  $th_{max}$ 에 따라  $Q_{avg}$ 가

달라지게 된다. 따라서 현재 라우터의 상황을 고려하지 못하기 때문에 혼잡상황에 유연하게 대처하지 못하는 단점이 있다. 또한,  $Q_{avg}$ 를 이용하여 혼잡제어를 효율적으로 하기 위해서는 큐로 들어오는 패킷을 모두 수용할 수 있는 충분한 크기의 큐가 필요하다. 하지만, 큐의 크기가 커지면 큐잉지연으로 인해 실시간 트래픽의 좋은 서비스가 어려우므로 작은 큐 크기를 유지하면서 성능을 개선할 수 있어야 한다.

## 2.2 ARED 알고리즘

ARED 알고리즘은 RED 알고리즘을 개선한 것으로, 라우터의 트래픽 부하에 따른 큐 길이 변화에 따라 패킷 제거 확률을 능동적으로 변화시켜 링크를 효율적으로 사용하고자 한 것이다.

그림 1은 ARED 알고리즘의 의사코드이다. 라우터의  $Q_{avg}$ 를 구한 후 그 길이를  $Q_{tar}$ (큐의 목표값)과 비교한다.  $Q_{avg}$ 가  $Q_{tar1}$ 보다 크고  $P_{max}$ 가 0.5이하이면,  $P_{max}$ 에  $\alpha$ 를 더해  $P_{max}$ 를 증가시켜 패킷 제거 확률을 높인다. 반면,  $Q_{avg}$ 가  $Q_{tar2}$ 보다 작고  $P_{max}$ 가 0.01보다 크면,  $P_{max}$ 에  $\beta$ 를 곱하여  $P_{max}$ 를 감소시켜 패킷 제거 확률을 낮춘다. 여기서,  $P_{max}$ 은 AIMD(Additive Increase Multiplicative Decrease) 알고리즘을 사용하며, 라우터의 트래픽 상황에 따라 패킷 제거 확률을 변화시켜  $Q_{avg}$ 를 결정한다.  $P_{max}$ 의 값은 0.01부터 0.5 범위 안에서 유지되며,  $Q_{avg}$ 의 변화에 패킷 제거 확률을

```

Initialization
   $Q_{avg} \leftarrow 0$ 
For each packet arrival
  Calculate average queue size  $Q_{avg}$ 
Every 0.5 seconds interval
  if ( $Q_{avg} > Q_{tar1}$  and  $P_{max} \leq 0.5$ )
     $P_{max} \leftarrow P_{max} + \alpha$ 
  elseif ( $Q_{avg} < Q_{tar2}$  and  $P_{max} \geq 0.01$ )
     $P_{max} \leftarrow P_{max} \cdot \beta$ 
  endif
Fixed parameters:
   $\alpha$ :  $\min(0.01, P_{max}/4)$ 
   $\beta$ : 0.9
   $Q_{tar1} = th_{min} + 0.6(th_{max} - th_{min})$ 
   $Q_{tar2} = th_{min} + 0.4(th_{max} - th_{min})$ 

```

그림 1. ARED 알고리즘의 의사코드

Fig. 1. Pseudo code for ARED algorithm.

천천히 변화시키므로 급격한 성능저하를 줄일 수 있다. 따라서,  $P_{max}$  를 결정하는  $Q_{tar}$  를 적절하게 설정하는 것이 중요하다.  $Q_{tar}$  은  $Q_{avg}$  를 유지하기 위한 목표값을 의미하고,  $\alpha$  와  $\beta$  는 각각  $P_{max}$  의 증가 및 감소요소이다. 이 알고리즘은 여러 트래픽 상황에 맞게 패킷 제거 확률을 변화시켜 평균 큐 길이를 안정화하는 것으로, 일정한 목표값에 평균 큐 길이를 유지시켜 RED 알고리즘보다 전체 성능을 향상시킨다. 또한, 순간적으로 발생하는 혼잡상황에 유연하게 대처할 수 있다. 하지만, 혼잡이 지속되면 실시간 트래픽에 대한 적응성이 떨어져 성능이 떨어질 수밖에 없다.

### 2.3 PQM 알고리즘

멀티미디어 데이터가 시간에 민감하고 프레임 유형별로 우선순위가 다른 특성을 이용하여, 큐를 관리하는 알고리즘이다.  $Q_{avg}$  가  $th_{min}$  보다 작으면, 수신 받은 패킷의 도달시간이 허용 가능한 지연 시간보다 작은 패킷을 큐에서 제거하여 불필요한 데이터의 전송을 방지하고 패킷수신을 위한 큐의 공간을 확보한다.  $Q_{avg}$  가  $th_{min}$  과  $th_{max}$  사이의 값이면, 해당 플로내의 패킷 중에서 수신된 패킷보다 낮은 우선순위를 갖는 패킷을 제거한다. 이것은 높은 우선순위의 패킷 보호로 멀티미디어 데이터의 품질을 향상시킬 수 있다.  $Q_{avg}$  가  $th_{max}$  보다 크면,  $Q_{avg}$  가  $th_{max}$  보다 작아질 때까지 큐에 도착하는 패킷을 제거한다. 이 알고리즘은 멀티미디어 트래픽의 특성을 고려하지만, 트래픽의 혼잡상황에 유연하게 대처하지 못하며 우선순위가 낮은 패킷의 손실률이 불필요하게 커지는 단점이 있다.

## III. 제안된 알고리즘

이 장에서는 네트워크의 혼잡정도를 고려하여 멀티미디어 스트림 전송을 효과적으로 하기 위한 APQM 알고리즘을 제안한다. 이 알고리즘은 ARED 알고리즘과 PQM 알고리즘의 단점을 보완한 것으로, 시간에 민감하고 프레임 유형별로 우선순위가 다른 특성을 가진 멀티미디어 데이터를 큐의 혼잡정도에 따라 효율적으로 관리하는 것을 목표로 한다. ARED 알고리즘은 큐 길이 변화에 따라 패킷 제거 확률을 다르게 하여 네트워크의 혼잡상황을 관리한다. 하지만, 시간에 민감한 멀티미디어 데이터의 경우 네트워크의 혼잡을 고려하여 패킷

을 제거한다 할지라도, 혼잡이 가중화되면 사용자가 만족할 만한 성능을 보장하기 어렵다. 따라서, 멀티미디어 데이터의 특성을 고려한 혼잡 제어가 필요하다. 반면, PQM 알고리즘은  $Q_{avg}$  가  $th_{min}$  이하인 비혼잡상황인 경우, 패킷의 지연시간을 비교하여 지연이 큰 패킷을 우선적으로 전송한다. 그리고  $Q_{avg}$  가  $th_{min}$  과  $th_{max}$  사이에 존재하는 혼잡상황인 경우, 패킷의 우선순위에 따라 패킷을 처리한다. 하지만 이러한 패킷처리는 기지국의 큐가 안정화된다고 할지라도  $Q_{avg}$  가  $th_{min}$  이하가 될 때까지 패킷을 우선순위에 따라 처리하므로 우선순위가 낮은 패킷의 손실률이 불필요하게 커지는 결과를 초래하게 된다. 따라서, 이러한 단점을 보완하고자  $Q_{avg}$  가  $th_{min}$  과  $th_{max}$  사이의 값인 혼잡상황인 경우  $Q_{avg}$  의 변화량에 따라 네트워크 혼잡정도를 판단하여 패킷을 처리하는 APQM 알고리즘을 제안한다.

이 알고리즘은 기지국의 큐가 비혼잡상황인 경우, 저장 공간이 충분하므로 RED 알고리즘과 같이 기지국에 들어오는 모든 패킷을 큐에 저장한다. 반면, 기지국의 큐가 혼잡상황인 경우,  $Q_{avg}$  의 변화량에 따라 혼잡정도를 파악하여 큐 안에 저장된 패킷을 제거한다. 이때,  $Q_{avg}$  의 변화량은 식 (4)를 이용하여 계산하며,  $Q_{var}$  로 나타낸다.  $Q_{avg\_pre}$  는 이전의 평균 큐 길이를 나타내고,  $Q_{avg}$  는 현재의 평균 큐 길이를 의미한다.

$$Q_{var} = Q_{avg} - Q_{avg\_pre} \quad (4)$$

$Q_{var}$  이 음의 값이면, 네트워크의 상태가 이전보다 원활해졌다고 판단하고, 기지국에서 수신 받은 프레임 패킷의 도달시간이 허용 가능한 지연 시간보다 작은 패킷을 큐에서 제거한다. 이때, 의존관계가 있는 프레임 패킷을 함께 제거하여, 비디오 재생에 불필요한 데이터의 전송을 방지하고 패킷 수신을 위한 큐 공간을 확보한다. 수신 받은 패킷의 허용 가능한 지연시간의 임계값은  $T_{th}$  로 나타내며, 식 (5)를 이용하여 계산한다. 식 (5)에서  $T_{GOP}$  는 각 GOP의 코딩 시간 간격이고,  $n$  은 1 GOP 내에서의 프레임 패킷 개수이다. 그리고  $N_{GOP}$  는 GOP의 번호이고,  $N_{VOP}$  는 프레임 패킷의 번호이다. 이 식은 비디오 스트림의 첫 번째 프레임 패킷이 서비스를 시작한 시점을 기준으로 각 프레임 패킷의 코딩 시간 간격을 계산하는 것으로, 큐에서 패킷을 제거하는 기준으로 사용한다.

$$T_{th} = \frac{T_{GOP}}{n} [n \times (N_{GOP} - 1) + (N_{VOP} - 1)] \quad (5)$$

멀티미디어 데이터는 지정된 시간 안에 수신 노드에 도달하지 않으면 비디오 재생에 사용할 수 없으므로 패킷 손실과 동일한 영향을 미치게 된다. 따라서, 기지국에서 수신한 프레임 패킷의 지연시간과  $T_{th}$ 를 비교한 후,  $T_{th}$ 보다 늦게 수신되면 프레임 패킷의 유형에 따라 차등적으로 처리한다. 이것은 손실된 프레임 패킷의 유형에 따라 다른 프레임 패킷에 미치는 영향이 다른 특성을 고려한 것이다. 멀티미디어 데이터의 I 프레임 패킷은 가장 중요한 데이터로 손실되면 해당 GOP 내의 비디오 데이터가 복원될 수 없다. 따라서, 임계값보다 늦게 수신된 I 프레임 패킷을 포함하는 해당 GOP의 모든 프레임 패킷을 큐에서 제거한다. P 프레임 패킷의 손실은 해당 프레임의 전후에 있는 B 프레임 패킷과 뒤따르는 해당 GOP 내의 모든 P 프레임 패킷에 영향을 미친다. 따라서 임계값보다 늦게 수신된 P 프레임 패킷과 해당 GOP 내에서 수신된 P 프레임 패킷 이후의 모든 프레임 패킷을 큐에서 제거한다. 그리고 B 프레임 패킷의 손실은 다른 프레임 패킷에 영향을 미치지 않으므로, 임계값보다 늦게 수신된 B 프레임 패킷만을 큐에서 제거한다. 반면,  $Q_{var}$ 이 양의 값이면, 기지국의 큐 혼잡이 이전보다 가중화된 것이므로 패킷의 유형을 고려한 제거확률로 패킷을 제거한다. 이는 중요도가 높은 패킷을 보호하기 위함이다. 패킷제거는 자신의 전후 프레임 패킷에 영향을 받는 MPEG의 특성을 고려하여, 큐에 저장된 패킷의 뒤쪽부터 제거한다. I 프레임 패킷이 기지국의 큐에 수신되면, 우선순위가 가장 낮은 B 프레임 패킷을 제거하고, B 프레임 패킷이 없으면 P 프레임 패킷을 제거하여 큐 공간을 확보한다. 그리고 P 프레임 패킷이 큐에 없으면 수신된 I 프레임 패킷을 제거한다. P 프레임 패킷이 큐에 수신되면, 큐에 있는 B 프레임 패킷을 제거하고, B 프레임 패킷이 없으면 수신된 P 프레임 패킷을 제거한다. 그리고 B 프레임 패킷이 수신된 경우, 수신된 B 프레임 패킷을 제거하여 우선순위가 높은 프레임 패킷을 수신하도록 한다.

$$th_{max} = th_{min} \cdot R(T(p_i^k)) \quad (6)$$

식 (6)은 프레임 패킷 유형에 따라 적용한  $th_{max}$ 로써, 식 (2)에 적용하여 사용한다. 이 식은  $th_{min}$ 과  $R(T(p_i^k))$ 을 이용하여  $th_{max}$ 를 설정한 것으로, RED

알고리즘에서 사용한  $th_{max}$  설정 방법을 이용한 것이다<sup>[2]</sup>.  $R(T(p_i^k))$ 는 데이터 패킷 유형에 따라 우선순위를 반영하기 위한 상수값으로,  $R(D) \gg R(B) \gg R(P) \gg R(I)$ 의 관계를 가지며, 값이 작을수록 패킷의 우선순위가 높다. 여기서  $p_i^k$ 는 플로  $i$ 의  $k$ 번째 패킷을 의미하고,  $T(p_i^k)$ 는 패킷의 종류를 나타낸다. 그리고 I, P, 그리고 B는 멀티미디어 데이터의 프레임 패킷을, D는 비멀티미디어 데이터를 나타낸다. 따라서, I 프레임 패킷은 가장 작은  $th_{max}$ 를 갖게 되어 패킷 제거 확률이 가장 낮고, 비멀티미디어 데이터 패킷은 가장 큰  $th_{max}$ 를 갖게 되어 가장 높은 패킷 제거 확률이 적용된다.

그림 2는 APQM 알고리즘의 의사코드이다. 기지국의 큐가 비어있는 경우,  $Q_{avg}$ 는 0,  $count$ 는 -1의 값을 갖고, 패킷이 큐에 도착하면  $Q_{avg}$ 를 계산하여, 임계값과 비교한 후 패킷을 처리한다.  $Q_{avg}$ 가  $th_{min}$ 보다 작으면, 기지국의 큐가 비혼잡상황이므로 수신된 패킷을 모두

1	Initialization
2	$Q_{avg} \leftarrow 0, count \leftarrow -1$
3	For each packet arrival
4	calculate average queue size $Q_{avg}$
5	$Q_{avg} = (1 - W_q)Q_{avg} + W_q \times Q_{size}$
6	<b>if</b> ( $Q_{avg} < th_{min}$ )
7	save packet in queue
8	<b>else if</b> ( $th_{min} < Q_{avg} < th_{max}$ )
9	$Q_{var} = Q_{avg} - Q_{avg\_pre}$
10	<b>if</b> ( $Q_{var} < 0$ )
11	<b>if</b> ( $d_i^k > T_{th}$ )
12	remove $p_i^k$
13	<b>else</b>
14	save packet in queue
15	<b>endif</b>
16	<b>else</b>
17	increment $count$
18	$th_{max} = th_{min} \cdot R(T(p_i^k))$
19	calculate probability $p_a$ :
20	$p_i^k$ is removed by $p_a$
21	$count \leftarrow 0$
22	<b>endif</b>
23	<b>else</b>
24	remove $p_i^k$
25	<b>endif</b>

그림 2. APQM 알고리즘(제안방법)의 의사코드  
Fig. 2. Pseudo code for APQM algorithm.  
(proposed method)

저장한다. 그리고  $count$  값은 큐가 혼잡한 상황에서 사용되는 파라미터이므로, -1로 유지한다. 반면,  $Q_{avg}$  가  $th_{min}$  과  $th_{max}$  사이의 값이면 큐가 혼잡한 상황이므로,  $Q_{avg}$  의 변화량에 따라 패킷 제거 확률을 다르게 적용하여 패킷을 제거한다.  $Q_{avg}$  의 변화량인  $Q_{var}$  이 음의 값이면, 패킷의 지연시간과 패킷의 허용 가능한 지연시간을 비교한다. 이때,  $d_i^k$  는 패킷의 지연시간을 나타내고,  $T_{th}$  는 식 (5)를 이용하여 계산한다.  $d_i^k$  가  $T_{th}$  보다 크면, 패킷을 큐에서 제거하고 그렇지 않으면 큐에 저장한다. 반면,  $Q_{var}$  이 양의 값인 경우, 큐에서 처리해야 할 트래픽이 많아진 것이므로, 큐에 도착하는 패킷 유형에 따라 패킷 제거 확률을 다르게 적용한다. 즉, 패킷의 우선순위에 따라 제거하는 것이다. 패킷이 큐에 들어올 때마다  $count$  값을 증가시키고, 우선순위가 높은 패킷이 작은  $th_{max}$  값을 갖게 하여, 낮은 패킷 제거 확률이 적용되도록 한다. 프레임 패킷마다 계산된  $th_{max}$  는 식 (2)의  $p_b$  에 적용되고, 이  $p_b$  를 식 (3)에 적용하여 최종 패킷 제거 확률인  $p_a$  를 계산한다. 패킷이  $p_a$  에 의해 제거되면,  $count$  값을 0으로 초기화한다.  $Q_{avg}$  가  $th_{max}$  보다 크면,  $Q_{avg}$  가  $th_{max}$  보다 작아질 때까지 큐에 도착하는 패킷을 제거한다. 그림 3은 제안한 알고리즘의 이해를 돕기 위한 순서도이다.

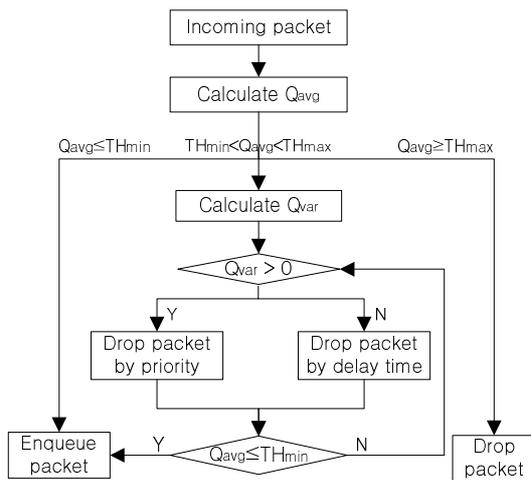


그림 3. APQM 알고리즘의 순서도  
Fig. 3. Flowchart for APQM algorithm.

#### IV. 성능 분석 및 평가

제안된 큐 관리 알고리즘의 성능 평가를 위하여

NS-2를 이용하여 환경을 구성하였다. 총 3대의 서버 중 한 대의 서버에서는 멀티미디어 데이터를 보내고, 나머지 두 서버는 각각 CBR 트래픽과 버스트 트래픽을 전송한다. 실제 인터넷과 유사한 환경에서 시뮬레이션 하기 위해 버스트 트래픽 서버와 CBR 서버, 그리고 Foreman 비디오 서버의 sending rate의 비율을 1: 0.1: 0.5로 하였다. 여기서 CBR 트래픽은 데이터율이 변하지 않고 고정된 것으로 평균 데이터율이 최대 데이터율과 동일하고, 버스트 트래픽은 데이터율이 아주 짧은 시간 동안 급변하게 되는 것을 말한다. 시뮬레이션에서 멀티미디어 데이터는 움직임이 많은 Foreman 비디오를 이용하였으며, RED, ARED, PQM, 그리고 제안한 알고리즘을 기지국의 큐에 적용하여 성능을 비교하였다.

표 1은 시뮬레이션에서 사용한 파라미터 값을 나타낸다. Queue size는 기지국에서 최대 수용할 수 있는 큐의 공간 크기를 의미하고,  $th_{min}$  은 제거 확률을 결정하는 큐의 최소 임계값을 나타낸다.  $th_{max}$  는 패킷 제거 확률을 결정하는 최대 임계값으로  $th_{min}$  의 4배한 값으로 사용하였으며,  $P_{max}$  는 패킷 제거를 결정하는 최대 확률값으로 1이다. 제안한 알고리즘과 PQM 알고리즘은 RED 알고리즘과 동일한 파라미터를 사용한다.  $th_{min}$  은 식 (7)을 이용하여 계산하고, 우선순위가 높을수록 큰 값을 갖게 된다.  $W_q$  는 실제 큐 크기가 변화가 얼마나 빨리 평균 큐 크기에 반영되는가를 결정하는 파라미터이다. 이 값이 너무 작으면 실제 큐 크기의 변화를 평균 큐 크기에 제대로 반영할 수 없고, 지나치게 크면 평균 큐 크기가 실제 큐 크기에 따라 과민하게 변화하여 순간적으로 발생하는 혼잡에 패킷을 제거하는 결과를 초래하게 된다. 그러므로 적절한 값을 설정하는 것이 중요하다. 그리고 ARED에서 사용하는  $\alpha$  와  $\beta$  는 각각  $P_{max}$  의 증가 및 감소요소이다.

표 1. 시뮬레이션 파라미터  
Table 1. Simulation parameters.

Simulation parameters	RED	ARED
Queue size	100	100
$th_{min}$	20	20
$th_{max}$	80	80
$P_{max}$	0.02	0.02
$W_q$	0.002	0.002
$\alpha$	.	0.01
$\beta$	.	0.9

$$th_{\min} = th_{\max} \cdot \frac{1}{R(T(p_i^k))} \quad (7)$$

4.1 수신단에서의 PSNR비교

멀티미디어 서비스에 대한 QoS는 일반적으로 이미지 복원의 품질 측정을 위해 사용되는 PSNR값을 비교하여 평가한다. PSNR은 식 (8)과 같이 MSE(Mean Squared Error)에 의해 정의되며, MSE는 식 (9)에 의해 정의된다. 멀티미디어 서버에서 전송한 원본 비디오 이미지는 RED, ARED, PQM 그리고 APQM 알고리즘이 적용된 기지국을 통해 수신단으로 전송된다. 멀티미디어 서버에서 전송한 원본 비디오 이미지와 수신단에서 받은 영상을 복원했을 때의 이미지 사이의 PSNR을 측정하여 각 알고리즘을 평가하였다. PSNR의 값이 클수록 원본에 가까운 것으로 QoS가 높다고 할 수 있다. 시뮬레이션 결과, 각 알고리즘을 적용했을 때 Foreman 비디오의 평균 PSNR값은 RED 알고리즘이 29.97, ARED 알고리즘은 32.92, PQM 알고리즘은 34.75이다. 그리고 APQM 알고리즘은 36.45로 측정되었다. APQM 알고리즘을 패킷의 특성을 고려하지 않는 RED 알고리즘과 비교했을 때 수신단에서 복원한 Foreman 비디오의 PSNR이 21.6% 개선되었고, 패킷의 특성 고려없이 기지국의 혼잡상황만을 고려한 ARED 알고리즘보다 10.7% 향상되었다. 그리고 멀티미디어 데이터의 특성을 고려한 PQM 알고리즘보다 4.9% 좋은 성능을 보인다. 따라서, 큐의 혼잡정도에 따라 패킷의 특성을 고려하여 패킷을 처리하는 APQM 알고리즘이 기존의 알고리즘보다 성능이 개선되었다.

$$PSNR = 10 \cdot \log_{10} \left( \frac{MAX_I^2}{MSE} \right) = 20 \cdot \log_{10} \left( \frac{MAX_I}{\sqrt{MSE}} \right) \quad (8)$$

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \| I(i,j) - K(i,j) \|^2 \quad (9)$$

그림 4는 기지국에 RED, ARED, PQM, 그리고 APQM 알고리즘을 적용하여 전송한 Foreman 비디오를 수신단에서 복원하였을 때, PSNR값의 차이를 보여 준다. GOP의 크기는 30 프레임으로 설정하여, 패킷 손실이 30 프레임이상 전파되지 않도록 하였다. 1 GOP에서는 패킷 손실이 없으므로, 모두 높은 PSNR값을 갖는다. 하지만, 트래픽의 많아지면서 혼잡상황이 발생하여

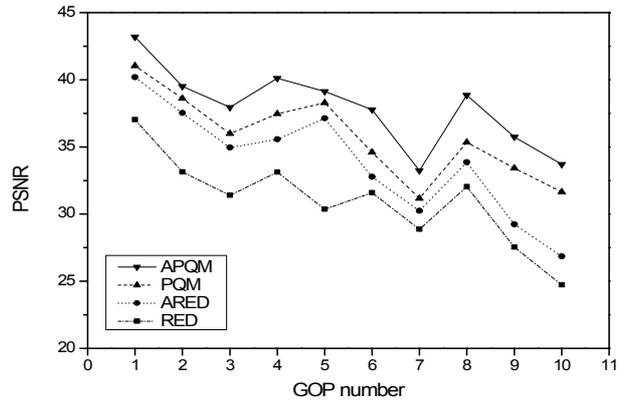


그림 4. 각 GOP별 평균 PSNR  
Fig. 4. Average PSNR versus GOP number.

PSNR값이 점점 감소한 것을 볼 수 있다. 혼잡상황 이후, 큐에 입력되는 패킷이 제거되어 큐의 혼잡상황이 다소 원활해져 PSNR값이 다시 증가하게 된다. 하지만, 다시 큐의 혼잡상황으로 PSNR값이 감소하게 된다. 그림 4를 통해 제안한 알고리즘이 다른 알고리즘보다 전체적으로 높은 PSNR값을 가져 멀티미디어 서비스 품질이 향상된 것을 알 수 있다.

4.2 전체 패킷 수신 확률

그림 5는 데이터의 송수신량으로 각 서버에서 전송한 패킷의 수신 확률을 비교한 것이다. 데이터의 수신량을 송신량으로 나눈 값이 1에 가까울수록 패킷 수신률이 높은 것이다. RED 알고리즘은 패킷 유형을 고려하지 않으므로, 트래픽이 고르게 수신되며 그중에서도 버스트 트래픽의 수신 확률이 조금 더 적다. ARED 알고리즘은 큐의 혼잡정도를 고려하여 패킷을 제거하므로 RED 알고리즘보다 멀티미디어 데이터의 수신 확률이

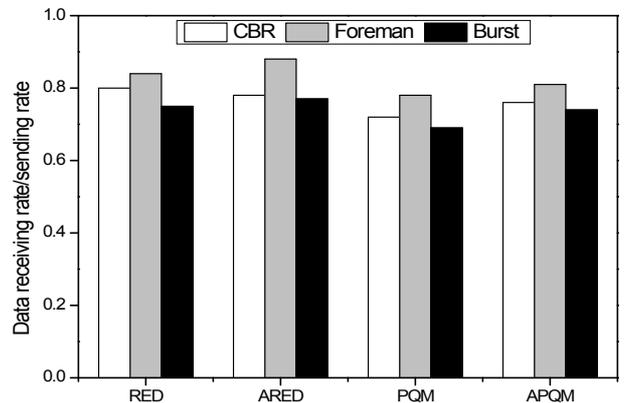


그림 5. 각 알고리즘의 패킷 수신 확률  
Fig. 5. Packet receiving probability of algorithms.

높다. PQM 알고리즘은 네트워크가 혼잡할 때 패킷의 우선순위에 의해 패킷을 제거하므로 다른 알고리즘보다 패킷 수신량이 적다. 그리고 APQM 알고리즘은 RED와 ARED 알고리즘보다는 멀티미디어 데이터의 수신률이 낮고, PQM 알고리즘보다는 수신률이 높으며 비멀티미디어 데이터의 수신률은 높다. 이것은 순간적으로 발생하는 혼잡상황에 유연하게 대처하고 멀티미디어 데이터의 특성을 고려하기 때문이다.

## V. 결 론

본 논문에서는 기지국의 혼잡정도에 따라 각 프레임 별로 우선순위가 차등적이고 시간에 민감한 멀티미디어 데이터의 특성을 이용하여 멀티미디어 서비스의 품질을 향상시키기 위한 APQM 알고리즘을 제안하였다. 이 알고리즘은 큐의 혼잡정도를 감지하여, 기지국의 큐가 혼잡한 상황에서 저장 공간이 충분해지면, 허용 가능한 지연 시간보다 늦게 도착한 패킷을 큐에서 제거하여 큐의 공간을 확보한다. 그리고 기지국의 큐가 혼잡해지면, 우선순위가 낮은 패킷을 제거하여 높은 우선순위의 패킷을 보호함으로써 멀티미디어 콘텐츠의 품질을 향상시키는 것이다. 성능평가를 위해 총 3대의 서버를 사용하였다. 각 서버는 멀티미디어 데이터, CBR 트래픽, 그리고 버스트 트래픽을 전송한다. 각 서버에서 보낸 데이터는 RED, ARED, PQM 그리고 APQM 알고리즘이 적용된 기지국을 통해 수신 노드로 전송된다. 수신단에서 복원한 영상의 PSNR값과 비디오의 원본 영상을 비교하여 각 알고리즘에 대한 멀티미디어 서비스의 품질을 평가하였다. APQM 알고리즘은 RED 알고리즘과 비교했을 때 Foreman 비디오의 PSNR이 21.6% 개선되었고, ARED 알고리즘보다 10.7%, PQM 알고리즘보다 4.9% 향상되었다. 제안한 알고리즘이 영상 재생에 불필요한 패킷을 제거하고 중요도가 높은 패킷을 수신함으로써 높은 질의 서비스를 제공하는 것을 확인하였다.

## 참 고 문 헌

- [1] [http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/VNI\\_Hyperconnectivity\\_WP.pdf](http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/VNI_Hyperconnectivity_WP.pdf)
- [2] K. S. Shin, W. O. Yoon, J. H. Jung, and S. B. Choi, "VOD Multicast Using CIWP and P2P Partial Stream Transfer", 13th International Multimedia Modeling Conference, MMM2007, Singapore, LNCS 4352, Part II, pp. 104-114, Jan. 2007.
- [3] S. Floyd and V. Jacobson, "Random Early Detection gateways for Congestion Avoidance", IEEE/ACM Trans. on Networking, pp. 397-413, Aug. 1993.
- [4] S. Floyd and R. Gummadi, and S. Shenker "Adaptive RED: An algorithm for Increasing the Robustness of RED's Active Queue Management", AT&T Center for Internet Research at ICSI, Aug. 2001.
- [5] "Advanced QoS Services for the Intelligent Internet", Cisco white paper, [http://www.cisco.com/warp/public/cc/pd/iosw/ioft/ioqo/tech/qos\\_wp.html](http://www.cisco.com/warp/public/cc/pd/iosw/ioft/ioqo/tech/qos_wp.html)
- [6] J. Orozco and D. Ros, "An Adaptive RIO(ARIO) Queue Management Algorithm", in Proceedings of QoSIS 2003, LNCS 2811, pp. 11-20, 2003.
- [7] W. Feng, D. D. Kandlur, D. Saha and K. G. Shin, "BLUE: A New Class of Active Queue Management Algorithms", Technical Report, UM CSE\_TR\_387\_99, University of Michigan, 1999.
- [8] R. Pan, B. Prabhakar, L. Breslau, and S. Shenker, "Approximate Fair Allocation of Link Bandwidth," IEEE Micro, vol. 23, no. 1, pp. 36-43, 2003.
- [9] A. Tang, J. Wang, and S. Low, "Understanding CHOKe: Throughput and Spatial Characteristics," IEEE/ACM Transactions on Networking, vol. 12, no. 4, pp. 694-707, Aug. 2004.
- [10] 김지원, 윤정희, 장은미, 최상방, "QoS 보장을 위한 멀티미디어 데이터 큐 관리 알고리즘 연구," 정보과학회논문지-시스템 및 이론, 제 38권 제3호, pp.123-131, 2011.06.
- [11] I. E. G. Richardson, H.264 and MPEG-4 video compression, Wiley, 2003.
- [12] Network Simulator(NS), University of California at Berkeley, CA, 1997. Available via <http://isi.edu/nsnam/ns>.
- [13] P. Ramanathan, and P. Agrwal, "Adapting Packet Fair Queuing Algorithms to Wireless Networks", ACM MOBICOM'98, Oct. 1998.

## 저 자 소 개



김 지 원(학생회원)  
2007년 건양대학교 전자정보  
공학과 학사 졸업.  
2009년 인하대학교 전자공학과  
석사 졸업.  
2010년~현재 인하대학교  
전자공학과 박사과정.

<주관심분야 : 멀티미디어 통신, 무선 통신, 네트  
워크 >



윤 정 희(학생회원)  
1998년 인하대학교 전자계산  
공학과 학사 졸업.  
2006년 인하대학교 정보컴퓨터  
교육학과 석사 졸업.  
2008년~현재 인하대학교  
전자공학과 박사과정.

<주관심분야 : 컴퓨터 구조, 무선통신, 네트워크>



장 은 미(학생회원)  
2002년 인하대학교 전자전기  
컴퓨터공학부 학사 졸업.  
2005년 인하대학교 정보컴퓨터 교  
육학과 석사 졸업.  
2010년~현재 인하대학교  
전자공학과 박사과정.

<주관심분야 : 컴퓨터 구조, 네트워크>



최 상 방(평생회원)  
1981년 한양대학교 전자공학과  
학사 졸업.  
1981년~1986년 LG 정보통신(주).  
1988년 University of washinton  
석사 졸업.

1990년 University of washinton 박사 졸업.  
1991년~현재 인하대학교 전자공학과 교수  
<주관심분야 : 컴퓨터 구조, 컴퓨터 네트워크, 무  
선 통신, 병렬 및 분산 처리 시스템>