DOI: 10.3745/KIPSTC.2008.15-C.6.523

제한된 재전송 횟수를 지원하는 SR-ARQ 프로토콜의 큐잉 지연 분석 모델

한제찬[†] · 김범준[††] · 이재용[†††]

요 약

본 논문은 제한된 재전송 횟수(retransmission persistence)를 갖는 SR-ARQ(Selective Repeat Automatic Repeat reQuest) 프로토콜의 큐잉 지연에 대한 분석 모델을 제안한다. SR-ARQ는 링크 계층 프로토콜로서 전송 중에 손실되거나 손상된 패킷을 재전송을 통해 복구하는 기능을 한다. SR-ARQ의 재전송 횟수는 SR-ARQ의 손실 복구 성능 뿐만 아니라 지연 성능에 큰 영향을 미치는 패러미터이다. 현재 최대 재전송 횟수가 SR-ARQ의 성능에 어떤 영향을 미치는 지에 대한 연구는 많지 않다. 이에, 본 논문에서는 M/G/1 큐잉 모델을 바탕으로 제한된 재전송 횟수를 갖는 SR-ARQ의 큐잉 모델을 제안하고 이를 통해 SR-ARQ의 큐잉 지연에 대한 재전송 횟수의 영향을 분석한다. 또한, OPNET을 이용한 모의실험을 통해 제안된 큐잉 모델의 정확성을 검증한다. 특히, 본 논문은 패킷 손실율과 트래픽 부하를 달리하는 여러 통신 환경에서 재전송 횟수가 SR-ARQ의 큐잉 지연 성능에 어떤 영향을 미치는 지를 수학적 분석 결과 및 모의실험 결과를 통해 명확하게 보여준다.

키워드 : SR-ARQ, 재전송 횟수, M/G/1 모델, 큐잉 지연

Queuing Analysis Model for the SR-ARQ Protocol with a Finite Retransmission Persistence

Jechan Han[†] · Beomjoon Kim[††] · Jaiyong Lee[†††]

ABSTRACT

In this paper, we analyze the mean queuing delay of selective-repeat automatic repeat request (SR-ARQ) protocol with the finite retransmission persistence. The retransmission persistence means the willingness of the protocol to retransmit a lost (or corrupted) packet to ensure reliable packet delivery across a lossy link. According to the retransmission persistence, SR-ARQ protocols have a different performance in terms of both packet delay and link reliability. So far, however, there is no serious study in the effect of the retransmission persistence on the SR-ARQ performance. We present a simple M/G/1 queuing model for the SR-ARQ protocol with the finite retransmission persistence by using the ideal SR-ARQ approximation. The mean queuing delay is obtained from the queuing model and verified its accuracy through the simulation results using the OPNET simulator. Both the analytical predictions and simulation results clearly show the effect of retransmission persistence on the queuing delay of the SR-ARQ protocol in various network conditions: packet loss rate and traffic condition over a wireless link.

Keywords : SR-ARQ, Retransmission Persistence, M/G/1 Model, Queuing Delay

1. Introduction

To ensure high reliability for data communications over the wireless link, many wireless access networks employ selective repeat automatic repeat request (SR-ARQ) as a link level error control scheme. Also, many studies have shown that SR-ARQ has the advantages of higher transport layer protocols such as transmission control protocol (TCP) [1,2]. In the literature, however, it has been shown that retransmissions by SR-ARQ cause highly variable and sometimes very high packet delay that may degrade the performance of higher layer protocols. RFC 3366 [1] described concretely the problems arising from the high and highly variable delay of

---

※ This research was supported by the MKE (Ministry of Knowledge Economy, Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment) (IITA-2008-C1090-0801-0038)

† 준 회 원 : 연세대학교 전기전자공학과 박사과정
†† 정 회 원 : 계명대학교 전자공학과 교수
††† 정 회 원 : 연세대학교 전기전자공학과 교수

논문접수 : 2008년 8월 26일
수 정 일 : 1차 2008년 9월 22일
심사완료 : 2008년 9월 24일

SR-ARQ. The excessive retransmissions by SR-ARQ may cause the interaction problem between TCP and SR-ARQ that is mainly represented by the spurious TCP retransmission timeout. Also, the additional delay introduced by SR-ARQ may be undesirable for delay-sensitive applications such as video on demand (VOD) and voice over IP (VoIP) [1].

Several previous studies have been carried out on the delay performance of SR-ARQ as well as its throughput [3-6]. In [3], Fantacci presented an analytical approach for analyzing the mean packet delay and the mean queue length at the transmitter based on the ideal SR-ARQ approximation. Rosberg and Shacham analyzed the re-sequencing delay and buffer occupancy at the re-sequencing buffer, assuming the heavy-traffic condition and a static radio channel [4]. In [5], Kim and Krunz developed a mean analysis for the total delay that a packet experiences in a SR-ARQ protocol, assuming the ideal SR-ARQ approximation and the heavy traffic condition. Also, they considered a time varying channel, a finite round-trip delay and a Markovian traffic source. [6] presented an analysis statistics on the delivery delay of a fully reliable SR-ARQ protocol assuming non-instantaneous feedbacks and the heavy traffic condition.

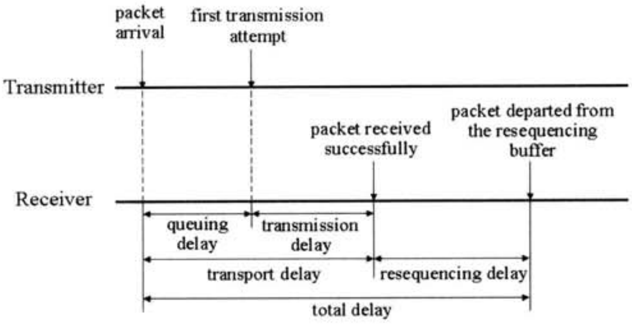
The existing analysis models, however, considered only the perfectly-persistence cases in which SR-ARQ protocols repeat retransmission of a lost (or corrupted) packet to infinity until the packet is successfully delivered. On the other hand, SR-ARQ protocols that are employed in real wireless access networks generally use the finite retransmission persistence that limits the maximum number of retransmission attempts for a lost packet [1,10,11]. Also, the existing queuing models did not definitely show the effects of the retransmission persistence on the queuing delay of the SR-ARQ protocol. In this paper we focus on developing a queuing mode for a SR-ARQ protocol with the finite retransmission persistence, as the first step of its delay performance analysis. In our queuing model, if all of the retransmission trials for a packet fail, the SR-ARQ protocol gives up recovering the lost packet by dropping it from the protocol, and moves on to forwarding subsequent buffered in-sequence packets. Our queuing model for the SR-ARQ protocol, concerned with the retransmission persistence as well as the packet loss rate on a wireless link and the traffic condition, provides a closed-form equation for the mean queuing delay.

The remainder of this paper is organized as follows. In Section 2, we briefly describe the SR-ARQ protocol and its delay components. Section 3 presents a queuing model for the SR-ARQ protocol with the finite retransmission persistence. In Section 4, we verify the accuracy of the analytical predictions about the queuing delay with simulation results and show the effect of the retransmission persistence on the queuing delay. Section 5 concludes this paper.

2. SR-ARQ Description

We consider an SR-ARQ protocol with the following features. Every packet has a unique sequence number in the SR-ARQ protocol. The transmitter sends packets consisting of payload and header according to their sequence number. A copy of each transmitted packet is temporarily kept in a buffer until the feedback message (acknowledgment (ACK) or negative ACK (NAK)) arrives. At the receiver side, based on the outcome of the error detection procedure and a check on a packet sequence number, feedback messages (ACK or NAK) are sent back to the transmitter. Once an ACK is received, the packet is removed from the transmitter. If a NAK arrives, the transmitter decides whether it will conduct a retransmission for a lost packet or not, based on its retransmission persistence. The retransmission persistence is defined as the willingness of the protocol to retransmit lost packet to ensure reliable packet delivery across a link [1]. In our paper, the SR-ARQ protocol uses the maximum number of retransmission attempts as the retransmission persistence. If the number of retransmission trials for a requested packet exceeds the predefined retransmission persistence, the SR-ARQ drops it from the protocol and moves on to forwarding subsequent buffered in-sequence packets. If a retransmission is permitted, the requested packet is inserted into the retransmission queue. A retransmission for a lost (or corrupted) packet is always prior to a transmission for new packets on a non-preemptive basis. Therefore, the transmitter cannot send a new packet until the retransmission queue becomes empty.

Generally, the overall packet delay of the SR-ARQ protocol consists of the queuing delay, the transmission delay and the resequencing delay as shown in (Fig. 1) [5]. When a packet arrives in the SR-ARQ transmitter, if there are other packets, it must wait in the transmission queue until its first transmission. The waiting time of a packet is the queuing delay. The second delay component is the time between the first transmission and the correct reception of the packet at the receiver. It is called the transmission delay. This mainly depends on the channel



(Fig. 1) The packet delay components of the SR-ARQ protocol [5]

behavior. The third delay component is the time spent in the receiver resequencing buffer. Generally, SR-ARQ protocols provide an ordered delivery of packets to the higher layer. Although a packet is received correctly, if there is a missing packet with a lower sequence number, the packet must wait in the resequencing queue until the missing packet is recovered. This time is the resequencing delay [5].

3. Analysis Model for the Queuing Delay

In this section, we introduce a queuing model for the SR-ARQ protocol with the finite retransmission persistence based on the work [5] that analyzes the performance of the SR-ARQ protocol on the basis of queuing theory. While [5] considers only the perfectly retransmission persistence case, our queuing model considers the finite retransmission persistence case that makes it possible to evaluate the effect of the retransmission persistence on the queuing delay of the SR-ARQ protocol.

3.1 Assumptions and Notations

Our model is based on an embedded Markov chain [5, 8] in which the number of packets in the queue is observed at the end of each time slot. For simplicity, we assume that the service time for a packet transmission is constant and the packet arrivals follow the Poisson process [7]. To get a simple closed-form equation for the mean queuing delay of the SR-ARQ protocol, a few assumptions are made as follows:

1) Time is slotted in a fixed length denoted by s (sec). A time slot corresponds to a single packet transmission.
2) The transmitter serves packets on first come first serve (FCFS) basis.
3) Packet losses on a wireless link occur following the Bernoulli process with the probability p ($0< p <1$).
4) A retransmission is always prior to a transmission of a new packet.
5) The receiver detects all of the lost (or corrupted) packets exactly, and immediately requests a retransmission for the lost (or corrupted) packet to the transmitter.
6) The transmitter has an infinite queue for new packets and retransmissions, and the receiver has an infinite queue for the ordered delivery of packets to higher protocols.
7) The ideal SR-ARQ approximation [5] is adopted; i.e. the SR-ARQ transmitter gets a feedback with no delay after a packet transmission [3, 5].

Key notations are summarized as follows:

1) Q[k]: The number of packets in the queue of the SR-ARQ transmitter at the end of the kth slot
2) A[k]: The number of new arrivals during the kth slot
3) V[k]: The number of packets departing from the SR-ARQ transmitter during the kth slot
4) μ: The service rate of the SR-ARQ transmitter (packets per second)
5) λ_{new}: The arrival rate of newly inserted packets into the SR-ARQ transmitter
6) λ_{rtx}: The arrival rate of packets retransmitted by the SR-ARQ transmitter
7) λ: The total arrival rate of packets in the SR-ARQ transmitter
8) r: The retransmission persistence, the maximum number of retransmission attempts
9) n: The total number of transmission attempts including retransmissions for a packet delivery.

3.2 Retransmission Probability

As mentioned before, the SR-ARQ protocol uses the maximum number of retransmission attempts for a packet as the retransmission persistence. In the case of the SR-ARQ protocol with the finite retransmission, the amount of retransmissions are limited by the value of retransmission persistence and closely depends on the queuing performance; mean queue length and mean queuing delay. To calculate the amount of retransmissions, we define the retransmission probability (p_{rtx}) that means the probability that when a packet loss is detected, the SR-ARQ transmitter retransmits it. Assuming an equilibrium state, we can calculated the value of p_{rtx} that let the SR-ARQ transmitter have the same average amount of retransmissions with the case using the maximum number of retransmissions attempts as the case.

First, when the SR-ARQ protocol decides whether it conducts a retransmission for a lost (or corrupted) packet or not, based on the retransmission persistence (r), the average number of packet transmission trials is

$$E[n]=\sum_{k=1}^{r+1} kP(n=k)=\frac{1-p^{r+1}}{1-p} \tag{1}$$

where n has the following distribution (reminding the assumption that packet losses follow the Bernoulli process).

$$P(n=k)=\begin{cases} p^{k-1}(1-p) & k=1,2,\dots,r \\ p^{r} & ,k=r+1\ . \end{cases} \tag{2}$$

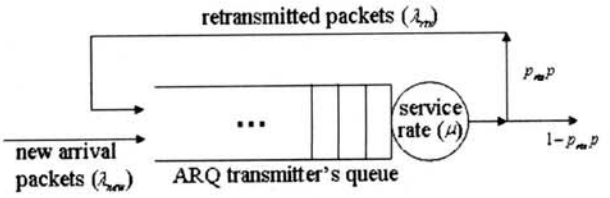
In the case that a retransmission of a lost packet is decided by the probability p_{rtx}, the average number of packet transmission trials is given by

$$E[n]=\sum_{k=1}^{\infty} k(p_{rtx}p)^{k-1}(1-p)=\frac{1}{1-p_{rtx}p}\ . \tag{3}$$

Substituting (1) into (3), we get the value of p_{rtx} corresponding to the value of r as follows.

$$p_{rtx}=\frac{1-p^{r}}{1-p^{r+1}} \tag{4}$$

Then, the SR-ARQ protocol is modeled by the queuing system with the retransmission probability p_{rtx}, as shown in (Fig. 2) The mean queuing delay of the SR-ARQ protocol depends on the arrival rate and service rate of the system. To obtain the mean queuing delay, therefore, it does not need to consider all the transmission history of a packet. At the equilibrium, we can approximate the total traffic load (ratio of the total arrival traffic including both new packet arrivals and retransmitted packets to the service rate) using the retransmission probability without loss of accuracy. We will show validity of the use of the retransmission probability through the simulation results in Section 4.



(Fig. 2) The network queuing model for the SR ARQ protocol

3.3 M/G/1 Queuing Model

The queue length at the end of the kth slot is determined by the queue length at the previous slot (Q[k-1]), the number of new arrivals at the kth slot (A[k]), and a packet transmission result at the kth slot (V[k]). Then, it is written as

$$Q[k]=\begin{cases} Q[k-1]-V[k]+A[k] & ,Q[k-1]>0 \\ A[k] & ,Q[k-1]=0\ . \end{cases} \tag{5}$$

Recalling the assumption that a time slot corresponds to a single packet transmission, V[k] has 0 or 1 according to a packet transmission result at the kth slot. V[k] has 1 in the following cases that a packet departs from the transmitter.

1) A packet is delivered to the receiver successfully.
2) A packet transmission fails, but the transmitter does not permit further retransmissions for the packet and drops it.

By the second case that V[k] has 1, the operation of the SR-ARQ protocol with the finite retransmission persistence is reflected in our queuing model. Though a packet loss is detected, the SR-ARQ protocol drops the packet rather than retransmits it, if the number of transmission attempts exceeds the predefined retransmission persistence (r). V[k] has 0 when there is no packet to be sent (Q[k-1]=0) or when a packet transmission fails at the kth slot and the packet is reinserted into the SR-ARQ transmitter for a retransmission. Consequently, V[k] is defined as a random variable that has 0 or 1 with the probabilities

$$P(V[k]=0)=\begin{cases} 1 & ,Q[k-1]=0 \\ p_{rtx}p & ,Q[k-1]>0 \end{cases} \tag{6}$$

$$P(V[k]=1)=\begin{cases} 0 & ,Q[k-1]=0 \\ 1-p_{rtx}p & ,Q[k-1]>0\ . \end{cases} \tag{7}$$

Next, denote

$$\hat{Q}[k]=Q[k-1]-V[k]\ . \tag{8}$$

Then, the queue length is expressed like this.

$$Q[k]=\hat{Q}[k]+A[k] \tag{9}$$

Since $\hat{Q}[k]$ and A[k] are independent, the probability generating function (PGF) of the queue length at the equilibrium state is given by

$$G_Q(z) = G_{\hat{Q}}(z) \times G_A(z) \tag{10}$$

where $G_{\hat{Q}}(z)$ and $G_A(z)$ denote the PGF of $\hat{Q}$ and the PGF of A, respectively. Then, $G_{\hat{Q}}(z)$ is

$$\begin{aligned} G_{\hat{Q}}(z) &= \sum_{k=0}^{\infty} P(\hat{Q}=k)z^k \\ &= P(\hat{Q}=0)z^0 + \sum_{k=1}^{\infty} P(\hat{Q}=k)z^k \end{aligned} \tag{11}$$

where

$$P(\hat{Q}=0) = P(Q=0) + P(Q=1, V=1) \tag{12}$$

$$P(\hat{Q}=k) = P(Q=k, V=0) + P(Q=k+1, V=1). \tag{13}$$

From (6), (7), and (11), $G_{\hat{Q}}(z)$ is given by

$$\begin{aligned} G_{\hat{Q}}(z) = &\frac{(z-1)(1-p_{rtx}p)P(Q=0)}{z} \\ &+ \frac{(1-p_{rtx}p + p_{rtx}pz)G_Q(z)}{z}. \end{aligned} \tag{14}$$

Next, reminding the assumption that the service time for a packet is constant and the packets arrive following the Poisson process, $G_A(z)$ is given by

$$G_A(z) = e^{\lambda_{new}s(z-1)}. \tag{15}$$

As a consequence, $G_{\hat{Q}}(z)$ is given by

$$G_Q(z) = \frac{(z-1)(1-p_{rtx}p)P(Q=0)}{z/G_A(z) - (1-p_{rtx}p + p_{rtx}pz)}. \tag{16}$$

In order to derive the unknown term, *P(Q=0)*, in (16) the Jackson's theorem [8] is applied. The packets injected to the SR-ARQ transmitter are divided into two types: the new arrivals from the Poisson process with λ_{new} and the arrivals for the retransmission as shown in (Fig. 2) The SR-ARQ transmitter is modeled by the open network queuing model [8], and λ_{rtx} and λ are calculated as follows.

$$\lambda_{rtx} = \lambda p_{rtx} p \tag{17}$$

$$\lambda = \lambda_{new} + \lambda_{rtx} = \frac{\lambda_{new}}{1-p_{rtx}p} \tag{18}$$

The total traffic load, which is identical with the probability that the server is busy, is obtained from (18) as

$$\rho = \frac{\lambda}{\mu} = \frac{\lambda_{new}}{(1-p_{rtx}p)\mu} = \frac{\rho_{new}}{1-p_{rtx}p} \tag{19}$$

where ρ and ρ_{new} denote the total traffic load and the input traffic load ($=\lambda_{new}/\mu$) of the SR-ARQ protocol, respectively. Since *P(Q=0)* implies that the SR-ARQ transmitter is idle, $P(Q=0)=1-\rho$ so that

$$P(Q=0) = 1-\rho = \frac{1-p_{rtx}p - p_{new}}{1-p_{rtx}p} \tag{20}$$

By substituting (15) and (20) to (16), $G_Q(z)$ is obtained in terms of all known quantities. It follows that the mean queue length of the SR-ARQ transmitter can be derived as

$$E[Q] = G'_Q(1) = \frac{\rho_{new}(2-\rho_{new})}{2(1-\rho_{new}-p_{rtx}p)} \tag{21}$$

where $G'_Q(1)$ denotes the first derivative of $G_Q(z)$ with respect to z evaluated in z=1. According to Little's formula [8], the mean queuing delay (E[T]), is given by

$$E[T] = E[Q]/\lambda = \frac{\rho_{new}(2-\rho_{new})}{2\lambda(1-\rho_{new}-p_{rtx}p)} \tag{22}$$

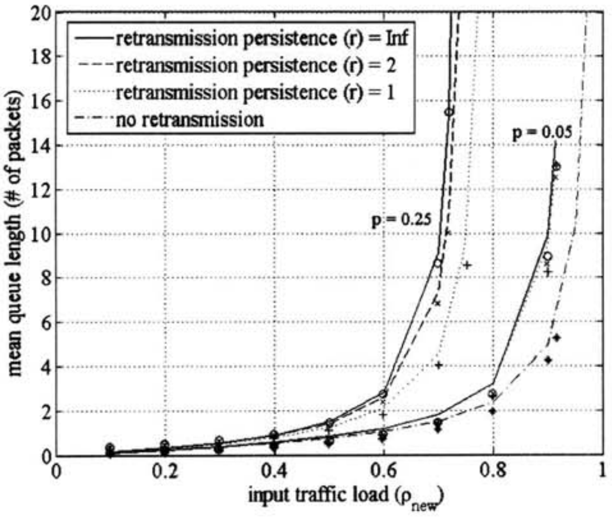
4. The Analysis and Simulation Results

The mean queuing delay statistics have been computed according to the above analysis model for various values of the packet loss rate, the input traffic load (ρ_{new}) into the SR-ARQ protocol, and the retransmission persistence. To test the accuracy, we performed simulations using the OPNET simulator [11] under the same condition with the analysis except that the ideal SR-ARQ approximation is not adopted in the simulations. In other words, the SR-ARQ transmitter in the simulation can receive a feedback message for a data packet transmission only after the round trip time, about 100 ms, over a wireless link. In <Table 1>, we have summarized the values of parameters used for simulations.

(Fig. 3) shows the mean queue length E[Q], obtained using (21), as a function of ρ_{new} for two values of p: 0.05 and 0.25. The figure includes the results about the SR-ARQ protocols with three different values of the retransmission persistence: 1, 2 and infinity (Inf). The mean queue length grows as new packet arrivals increase and starts increasing very rapidly when the input traffic

〈Table 1〉 Simulation Parameters

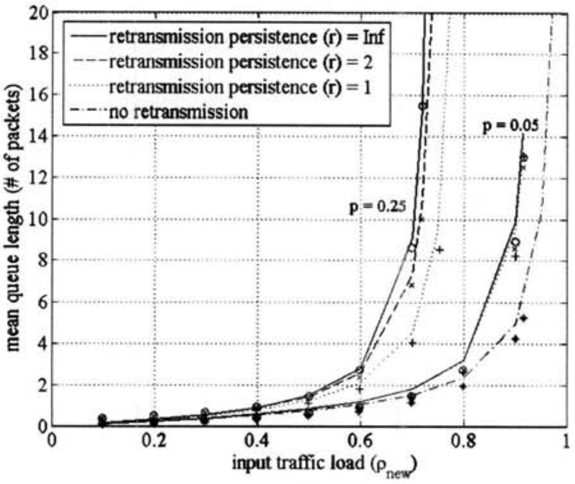
Item	Value
input traffic load (ρ_{new})	0.1~0.9
Service rate of SR-ARQ	100 packets/sec
Packet size of SR-ARQ	300 bytes
Retransmission persistence (r)	0, 1, 2, and Inf
Transmission queue size	Infinity
Retransmission queue size	Infinity
Retransmission timer of SR-ARQ	200 msec
Round-trip time (RTT)	100 msec
Packet loss rate (p)	0.05~0.25



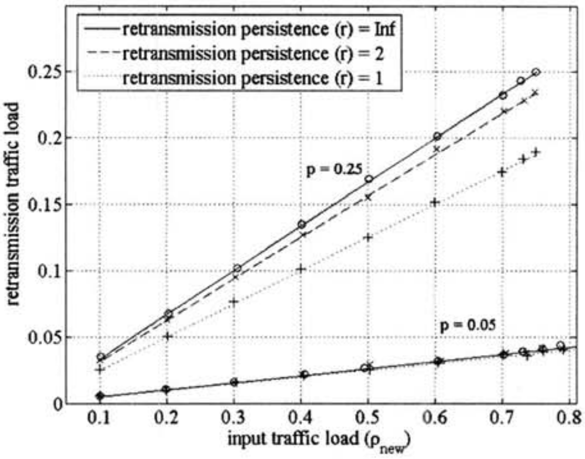
(Fig. 3) Mean queue length as a function of ρ_{new}, for p=0.05 and 0.25; analysis results (line), simulation results (marker)

load reaches a specific value which is different according to p and r. When p = 0.25, the queue length of SR-ARQ with the retransmission persistence Inf increases most quickly around input traffic load 0.65. And the cases with the retransmission persistence 2 and 1 start a sharp increasing of the mean queue length around input traffic load 0.7 and 0.75, respectively. On the other hand, all the cases show a similar result when p is 0.05. It is because most of packets are delivered successfully to the receiver through one or two times of transmissions. The case of no retransmission shows the results of SR-ARQ that does not allow any retransmission. The mean queue length increases rapidly at the input traffic load more than 0.9.

(Fig. 4) shows the mean queuing delay in the same simulation with (Fig. 3) The mean queuing delay has the same pattern with the mean queue length. We can observe that the mean queuing delay is closely related to the retransmission traffic load as well as the input traffic load. Even when the input traffic load is not high, an abrupt increment of the mean queuing delay occurs due to high retransmission traffic load. As shown in (Fig. 5), the retransmission traffic load is proportional to the input traffic load and the packet loss rate, and limited by the retransmission persistence. And, the difference in the retransmission traffic load according to the value of r becomes noticeable at high packet loss rate. In the high packet loss case (p=0.25), the input traffic load is not much (under 0.7) but, the retransmission traffic load grows above 0.2 that makes the total traffic load approach the leading point that results in an abrupt



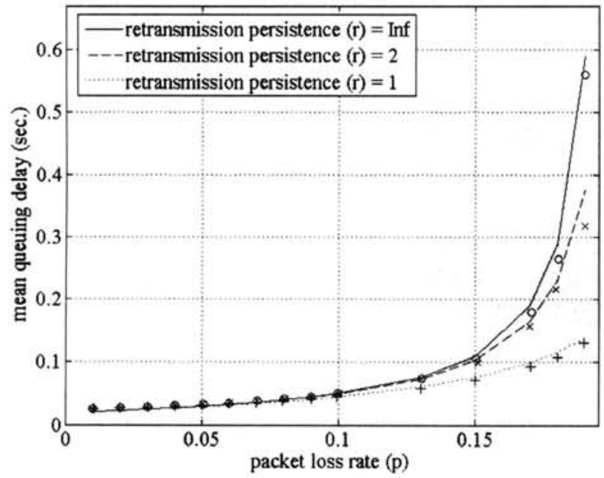
(Fig. 4) Mean queuing delay as a function of ρ_{new}, for p=0.05 and 0.25; analysis results (line), simulation results (marker)



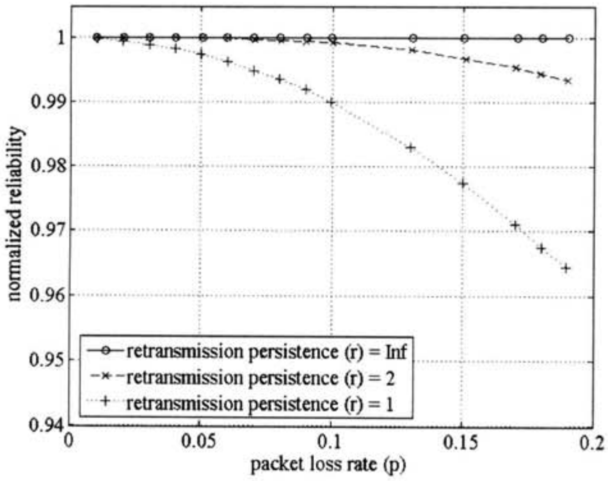
(Fig. 5) Retransmission traffic load as a function of ρ_{new}, for p=0.05 and 0.25; analysis results (line), simulation results (marker) increment of queuing delay. That means that excessive retransmissions of the SR-ARQ protocol can cause a sharp increase of queuing delay even though the input traffic load is not much. On the other hand, when the packet loss rate is low as 0.05, the amount of retransmissions is almost same in all SR-ARQ protocols regardless of the retransmission persistence because most packet losses are recovered by the first retransmissions. That is, in the low packet loss rate, 0.05, the retransmission traffic load stays under 0.05.

As shown in (22), the queuing delay closely depends on the amount of retransmissions as well as the amount of new packet arrivals. (Fig. 3) and (Fig. 4) show that there is a good agreement between the analytical results obtained by the proposed queuing model and simulation results. Particularly, as shown in (Fig. 5), the retransmission traffic load calculated by (17) shows also good agreement with the simulation results.

(Fig. 6) shows the mean queuing delay, as a function of p in the high traffic condition (ρ_{new}=0.8). The input traffic load was fixed in this scenario. Since the retransmission traffic load increases in proportion to the packet loss rate, however, the mean queuing delay also increases with p increasing. When p is more than 0.17, the mean queuing delay starts increasing very rapidly in two cases, retransmission persistence Inf and 2. On the other hand, SR-ARQ allowing only one retransmission keeps a low queuing delay as shown in (Fig. 6) However, SR-ARQ with low retransmission persistence shows poor performance in the aspect of the normalized reliability that is defined as the ratio of the number of successful packet deliveries to the number of total packet deliveries, compared to the other cases with the retransmission persistence Inf and 2, as shown in (Fig. 7)



(Fig. 6) Mean queuing delay as a function of p, for ρ_{new} = 0.8; analysis results (line), simulation results (marker)



(Fig. 7) Normalized reliability as a function of p, for ρ_{new} = 0.8

5. Conclusion

Our analysis model for the mean queuing delay of the SR-ARQ protocol has considered the finite retransmission persistence. The accuracy of the obtained analytical predictions has been verified through comparison with simulation results. Both analytical predictions and simulation results showed that the mean queuing delay closely depends on the retransmission traffic load as well as the input traffic load. Particularly, even though the input traffic load is not high, SR-ARQ protocols experience an abrupt increment of queuing delay when the total traffic load is more than 0.9 due to increase of retransmission traffic load in the high packet loss rate. Accordingly, our further research should be directed at developing an advanced SR-ARQ protocol that provides highly reliable packet transmission without such an abrupt increment of queuing delay as shown in our analysis results.

References

[1] G. Fairhurst and L. Wood, "Advice to Link Designers on Link Automatic Repeat reQuest (ARQ)," IETF RFC 3366, Aug., 2000.

[2] P. Karn et al., "Advice for Internet Subnetwork Designers," IETF RFC3819, June, 2004.

[3] Ramano Fantacci, "Queuing Analysis of the Selective Repeat Automatic Repeat Request Protocol Wireless Packet Networks," *IEEE Trans. Vehic. Tech.*, Vol.5, No.2, May, 1996.

[4] Z. Rosberg and N. Shacham, "Resequencing delay and buffer occupancy under the selective-repeat ARQ," *IEEE Trans. Inform. Theory*, Vol.35, pp.166-173, Jan., 1989.

[5] Jeong Geun Kim and Marwan M. Krunz, "Delay Analysis of Selective Repeat ARQ for a Markovian Source Over a Wireless Channel," *IEEE Trans. Vehic. Tech.*, Vol.49, No.5, Sept., 2000.

[6] Michele Rossi, Leonardo Badia and Michele Zorzi, "SR ARQ Delay Statistics on N-State Markov Channels with Non-Instantaneous Feedback," *IEEE Trans. Wireless Comun.*, Vol.5, No.6, June, 2006.

[7] Sheldon Ross, *Stochastic Processes*, second edition, John Wiley & Sons, New York, 1996.

[8] L. Kleinrock, *Queuing Systems*, volume I: Theory, John Wiley & Sons, New York, 1975.

[9] 3GPP TS 25.333, "Radio Link Control (RLC) Protocol Specification," v.6.4.0., June, 2005.

[10] 3GPP2~C.S0017-010-A, "Data Service Options for Spread Spectrum Systems: Radio Link Protocol Type 3," version 2.0, Sept., 2005.

[11] http://www.opnet.com/products/opnet-products

한제찬

e-mail : hjcyy@yonsei.ac.kr
2002년 연세대학교 기계전자공학부(학사)
2004년 연세대학교 전기전자공학과(공학석사)
2004년~현 재 연세대학교 전기전자공학과 박사과정
관심분야 : TCP congestion control, link level error control, Wireless TCP, and analysis modeling for ARQ

김범준

e-mail : bkim@kmu.ac.kr
2003년 연세대학교 전기전자공학과(공학박사)
2004년~2006년 LG전자 선임연구원
2006년~현 재 계명대학교 전자공학과 교수
관심분야 : IEEE 802.16 Wireless MAN, TCP congestion control, TCP interaction with ARQ

이재용

e-mail : jyl@yonsei.ac.kr
1977년 연세대학교 전자공학과(학사)
1984년 IOWA State Univ.(공학석사)
1987년 Iowa State Univ.(공학박사)
1987년~1994년 포항공과대학교 교수
1994년~현 재 연세대학교 전기전자공학과 교수
관심분야 : protocol design for wired/wireless QoS management, ubiquitous sensor network, and wireless multimedia support protocol