Gap Processing Time Analysis of Stall Avoidance Schemes for High-Speed Downlink Packet Access with Parallel HARQ Mechanisms

Li-Chun Wang and Chih-Wen Chang

Abstract—The parallel multichannel stop-and-wait (SAW) hybrid automatic repeat request (HARQ) mechanism is one of key technologies for high-speed downlink packet access in the wideband code division multiple access system. However, this parallel HARQ mechanism may encounter a serious *stall problem*, resulting from the error of the negative acknowledgement (NACK) changing to the acknowledgement (ACK) in the control channel. In the stall situation, the receiver waits for a packet that will be no longer be sent by the transmitter and stops delivering the medium access control (MAC) layer packets to the upper layer. The stall issue seriously degrades the quality of service for the high-speed mobile terminal owing to the high probability of NACK-to-ACK errors. In this paper, we present an analytical approach to compare three stall avoidance schemes: the timer-based, the window-based, and the indicator-based schemes. To this end, we first propose a new performance metric-gap processing time, which is defined as the duration for a nonrecoverable gap appearing in the MAC layer reordering buffer until it is recognized. Second, we derive the probability mass functions and the closed-form expressions for the average gap processing time of these three stall avoidance schemes. It will be shown that our analytical results match the simulations well. Further, by analysis, we demonstrate that the indicator-based stall avoidance scheme outperforms the timer-based and the window-based schemes. The developed analytical approaches can help determine a proper number of processes for the parallel SAW HARQ mechanisms. We also show that the analytical formulas can be used to design the number of acceptable fully loaded users for an admission control policy subject to the gap processing time constraint. In the future, our analysis can facilitate the MAC/radio link control (RLC) cross-layer design because the gap processing time in the MAC layer is closely related to the RLC timeout mechanism and the window size in the RLC retransmission mechanism.

Index Terms—HSDPA, stall, stall avoidance, HARQ, multichannel SAW HARQ, gap processing time.

1 Introduction

High-speed downlink packet access (HSDPA) has become an important feature for the wideband code division multiple access (WCDMA) system [1]. The HSDPA in the WCDMA system aims to deliver mobile data services at rates up to 10 Mbits/sec [2], [3]. The key enabling technologies for HSDPA include physical layer fast adaptive modulation and coding [4], [5], [6], fast packet scheduling in the medium access control (MAC) layer [7], [8], [9], [10], [11], fast cell selection [12], multiple input multiple output (MIMO) antenna [13], [14], and buffer overflow control [15].

In this paper, we investigate the stall avoidance techniques to enhance the MAC layer performance of a parallel multichannel stop-and-wait (SAW) hybrid automatic repeat request (HARQ) mechanism adopted in HSDPA [16], [17], [18], [19], [20], [21]. In such a fast HARQ mechanism, a reordering buffer is equipped at the receive entity because packets may arrive out of sequence. However, due to transmission errors in a wireless channel, the negative acknowledgement (NACK) control signal for a damaged/lost packet is likely changed to the acknowledgement (ACK) signal. In this situation, the transmitter mistakenly believes that the packet has successfully reached the destination. Meanwhile, the receiver keeps waiting for a packet which will not be sent again by the MAC layer retransmission scheme. We call this problem the *stall issue* if the reordering buffer has a nonrecoverable gap due to a NACK-to-ACK error. The stall of delivering the MAC layer data to the upper layer will delay the inevitable upper layer radio link control (RLC) retransmission [22], [23]. It has been reported that the probability of the NACK signal becoming the ACK signal can be as high as 10^{-2} for a high-speed mobile during handoff [24], [25]. Thus, resolving the stall problem becomes an important task to reduce the transmission delay for the HSDPA [26].

To resolve the stall issue for the multichannel SAW HARQ, there are two main research directions in the literature. The first direction is to improve the reliability of control packets by increasing the power of ACK or NACK signals [24]. The second direction is to design stall avoidance schemes to inform the receiver to stop waiting for the lost MAC layer packets and start forwarding all the received in-sequence packets to the upper layers [26], [27] [28], [29], [30]. Since the lost packet cannot be recovered by the MAC layer retransmission mechanism, the upper layer protocols will be responsible for requesting the retransmission of the lost MAC layer packets. In [27], a timer-based stall avoidance scheme was suggested to trigger the process

- L.-C. Wang is with the Department of Communication Engineering, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 300, Taiwan, ROC. E-mail: lichun@cc.nctu.edu.tw.
- C.-W. Chang is with the Institute of Computer and Communication Engineering, National Cheng Kung University, 1 Ta Hsueh Road, Tainan 701, Taiwan, ROC. E-mail: cwchang@ee.ncku.edu.tw.

Manuscript received 24 Mar. 2005; revised 21 Nov. 2005; accepted 19 Jan. 2006; published online 15 Sept. 2006.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-0075-0305.

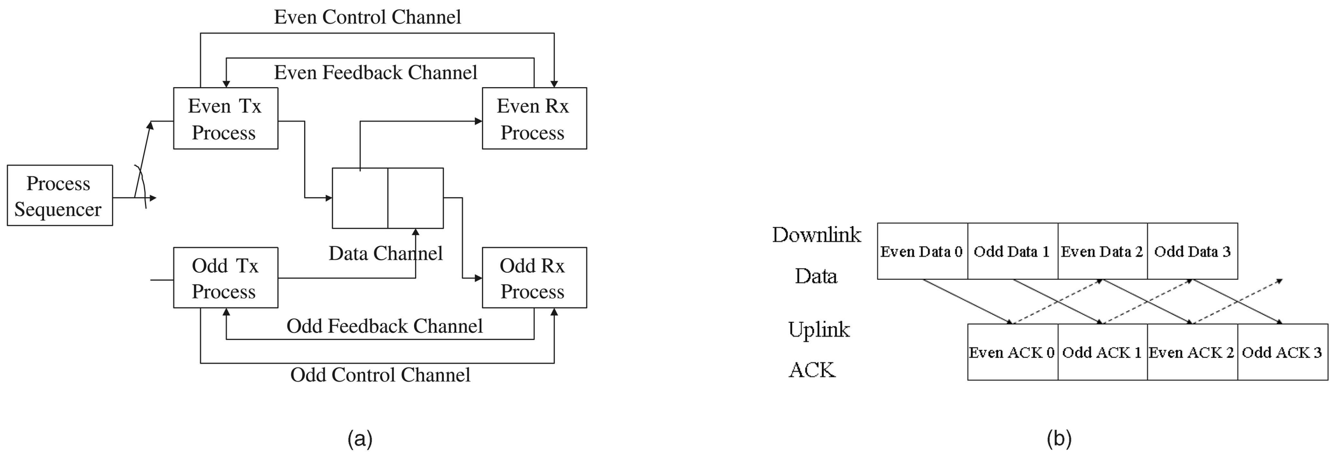

Fig. 1. The structure and timeline of the dual-channel SAW H-ARQ mechanism. (a) Structure. (b) Timeline.

of forwarding received packets to the upper layer as long as a gap of the received packets' sequence in the reordering buffer lasts over a predetermined expiration period. In [28], a window-based stall avoidance scheme proposed using a sliding window method to detect the stall situation in the reordering buffer before the timer expires. In [26], [29], [30], an indicator-based stall avoidance scheme is proposed with the aid of a new data indicator (NDI). This scheme recognizes the stall situation by checking the NDI information and the transmission sequence number (TSN) of each packet. If it is found that all the HARQ processes, instead of sending the expected missing packet, are transmitting either new packets or other old packets, the stall situation is confirmed. To our knowledge, the performances of the above stall avoidance schemes were only evaluated by extensive simulations [16], [22], [23].

The objective of this paper is to develop analytical methods to evaluate the performances of these three stall avoidance methods: the timer-based, the window-based, and the indicator-based schemes. To characterize the performances of the stall avoidance schemes, a new performance metric, called the gap processing time (defined in Section 3), is introduced in this paper. We derive the probability mass functions and the closed-form expressions for the average value of gap processing time of the three considered stall avoidance techniques. By simulations and analyses, we find that the indicator-based stall avoidance scheme significantly reduces the gap processing time compared to the timer-based and the window-based schemes. When applying these three stall avoidance schemes, the presented analytical approach can provide important information for determining a proper number of processes in the parallel SAW HARQ mechanism. It can also be used to design the allowable retransmissions for the timer-based and the window-based schemes. Furthermore, the number of acceptable fully loaded users can also be designed by the proposed analytical approach when the admission control and the gap processing time are jointly considered.

The rest of this paper is organized as follows: In Section 2, we describe the stall issue in the multichannel SAW HARQ mechanism. Section 3 introduces three kinds of stall avoidance schemes (the timer-based, the window-based, and the indicator-based schemes). In Section 4, we define a new performance metric-gap processing time and describe system assumptions. Sections 5, 6, and 7 derive the average gap processing time of the timer-based, the window-based, and the indicator-based stall avoidance schemes, respectively. Section 8 shows the performance of the stall resolution schemes in the Rayleigh fading channel. We also discuss the design principles of the expiration period for the timer-based scheme, the window size for the window-based scheme, and the number of parallel HARQ processes for the indicator-based scheme. Section 9 gives our concluding remarks.

2 Background

2.1 Multichannel SAW HARQ Mechanism

The multichannel SAW HARQ is adopted in the HSDPA system [1]. The basic idea of the multichannel SAW HARQ is to *"keep the data pipe full."* Fig. 1a illustrates an example of a dual-channel SAW HARQ consisting of an even transmitter and an odd transmitter [31]. As shown in Fig. 1b, after sending packet 0, the even transmitter waits for the acknowledgement from the receiver. Meanwhile, the odd transmitter starts sending packet 1. With two transmitters sending data alternatively, the dual-channel SAW HARQ can fully utilize the channel capacity, thereby achieving high throughput.

2.2 Definition of Type-I and Type-II Gaps

In this paper, a *gap* is defined as an idle space reserved for a lost packet in the reordering buffer of the receiver. We can further classify two types of gaps for the HARQ retransmission scheme. *Type-I gap* is defined as the lost packet that is possibly recovered in future retransmissions, while *Type-II gap* is the one that will never be sent again by the MAC retransmission scheme due to a NACK-to-ACK error. Whenever a Type-II gap appears in the reordering buffer, the process of sending packets to the upper layer is stalled. Note that a regular HARQ process usually cannot distinguish a Type-II gap from a Type-I gap. Thus, it is necessary to design a stall avoidance scheme to detect the occurrence of a Type-II gap to expedite data forwarding to the RLC layer. After a Type-II gap is detected, the available packets in

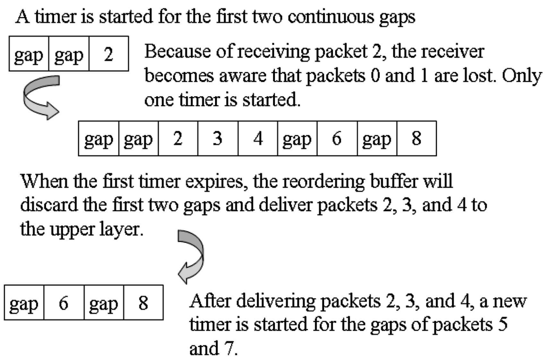

Fig. 2. An example of the timer-based stall avoidance scheme.

the reordering buffer as well as this Type-II gap must be flushed out to the RLC layer. Next, an RLC retransmission request is initiated for the missing Type-II gap [22], [29], [30].

3 The Stall Avoidance Schemes

Because a NACK control signal may be corrupted during transmissions in a wireless channel, many stall avoidance schemes are proposed to prevent a receiver from waiting for a missing packet due to a NACK-to-ACK error. Not only can the stall avoidance schemes detect the Type-II gap, but they also forward the already received packets at the MAC layer to the upper layer before triggering the RLC retransmission. Thus, compared to the purely RLC retransmission, the RLC transmission with the stall avoidance mechanism can expedite the delivery of the data by avoiding retransmitting the already received packets. In this section, we discuss three current stall avoidance schemes, the timer-based [27], the window-based [28], and the indicator-based schemes [26], [29], [30].

3.1 Timer-Based Scheme

The basic principles of the timer-based method in [27] are described as follows:

1. A timer is triggered when either a Type-I gap or a Type-II gap appears in the reordering buffer of the receiver.
2. As the timer expires, the receiver stops waiting for the lost packet and judges that the lost packet belongs to a Type-II gap.
3. The timer is reset if the missing packet is successfully retransmitted before the timer expires.

Fig. 2 illustrates the operation principles of the timer-based stall avoidance scheme. When packet 2 successfully arrives at the receiver, it is aware that packets 0 and 1 are missing and, thus, a timer is triggered. Note that only one timer is triggered for the consecutive gaps. Assume that packets 3, 4, 6, and 8 reach the receiver, but packets 5 and 7 are damaged. Based on the timer-bases stall avoidance scheme, a new timer will not be initiated for additional gaps until the previous timer expires. Thus, in this example, only after the timer set for packets 0 and 1 expires, a new timer will be set for the missing packets 5 and 7. After that, the receiver sends packets $2 \sim 4$ to the upper layer and requests for retransmitting packets 0 and 1 in the RLC layer.

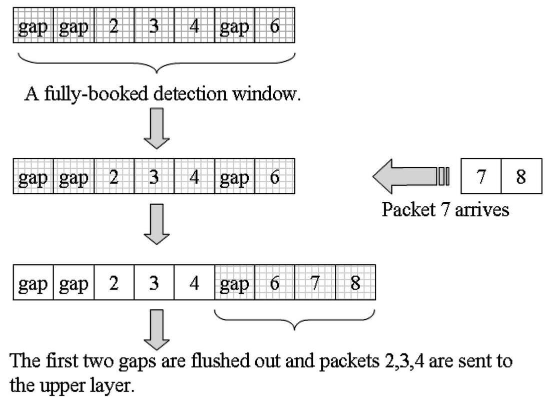

Fig. 3. An example of the window-based stall avoidance scheme with the detection window size equal to seven.

3.2 Window-Based Scheme

In [28], the window-based method was suggested to detect Type-II gaps in the reordering queue on top of the timer-based scheme. To explain the operation of the window-based stall avoidance scheme, we define the *detection window*. The detection window means a set of packets that are expected to arrive at the receiver. The detection window functions as a sliding window protocol. If a gap occurs, the detection window is initiated and two possible scenarios follow. For a Type-I gap, the detection window will shrink from the trailing edge after the gap is filled. For a Type-II gap, however, the trailing edge of the detection window is halted. Thus, the detection window expands from the leading edge when other subsequent packets arrive. In the long run, a predetermined maximum threshold of the detection window size will be reached.

A fully-booked detection window means that all available slots/seats in the reordering buffer have been taken. The detection window size is usually designed large enough to guarantee a successful retransmission. Thus, the gaps halting the trailing edge of a fully booked detection window usually belong to Type-II gaps. As a consequence, when a new packet arrives and the detection window is fully booked, the window-based stall avoidance scheme forwards Type-II gaps and subsequent received in-sequence packets in the reorder buffer to the upper layer in regardless of timer expiration.

Fig. 3 shows an example to illustrate the operation principles of the window-based stall avoidance scheme. Consider a detection window with a size of seven, which currently contains packets 2, 3, 4, and 6 and the gaps for missing packets 0, 1, and 5. Suppose that packets 0, 1, and 5 are Type-II gaps and new packets 7 and 8 arrive at this moment. Since the detection window is already *fully-booked* and its trailing edge is halted by packets 0 and 1, the window-based stall avoidance scheme can recognize that packets 0 and 1 are Type-II gaps. Accordingly, the receiver forwards packets 2, 3, and 4 together with gaps of packets 0 and 1 to the upper layer. Now, the detection window size becomes two and its trailing edge slides until the gap for packet 5. With more space available in the reordering buffer, packets 7 and 8 are accommodated and the detection window is reserved for packets 5 to 8.

TABLE 1
An Example of the Statuses in a 4-Process SAW HARQ Mechanism for a Type-II Gap Being Detected by Receiving New Packets in Three Processes and Old Packet in One Process

	Process ID in 4-process SAW HARQ			
	Process 1 (TSN,S_c,NDI)	Process 2 (TSN,S_c,NDI)	Process 3 (TSN,S_c,NDI)	Process 4 (TSN,S_c,NDI)
Cycle 1	(0,N→A,NEW)	(1,NACK,NEW)	(2,NACK,NEW)	(3,ACK,NEW)
Cycle 2	(4,ACK,NEW)	(1,ACK,OLD)	(2,ACK,OLD)	

3.3 Indicator-Based Scheme

In [26], [29], [30], a new stall avoidance scheme is proposed by taking advantage of the new data indicator (NDI). The NDI (just a one-bit tag) is associated with every packet and is transmitted in the high-speed downlink control channel [32]. The NDI is toggled for a new transmitted packet, but is not changed for a retransmitted old packet. By checking the NDI in the control channel and the transmission sequence number (TSN) in the traffic channel for each packet, the indicator-based stall avoidance scheme can determine whether the missing packet will be transmitted or not. If all HARQ processes are transmitting new packets or other old packets except for the expected missing packet, then this gap in the reordering buffer can be judged to be a Type-II gap.

Table 1 illustrates the status of a 4-process SAW HARQ mechanism for a particular user in two cycles. The cycle duration is defined as the sum of the transmission time interval (TTIs) with all different parallel processes transmitting one packet, e.g., the cycle duration is 4 TTIs in this case. The field in the table is filled with a triplet variable (TSN, S_c, NDI). Here, $S_c \in \{ACK, NACK, \text{and } N \rightarrow A\}$ denotes one of the three events: "receiving an ACK," "receiving a $NACK$," and "having a NACK-to-ACK error," respectively. The status NEW or OLD in the field NDI is equivalent to that the NDI is $changed$ for a new packet or is $unchanged$ for a retransmitted old packet, respectively. An empty field implies that this time slot is idle or assigned to other users. Assume that the target user requests to transmit five packets with $\mathrm{TSN} = 0 \sim 4$ from the RLC layer. The functions of the indicator-based stall avoidance scheme associated with Table 1 are explained as follows:

1. In cycle 1, the four parallel processes transmit packets 0 to 3, respectively. Assume that packet 3 passes the cyclic redundancy check (CRC), but packets 0 $\sim$ 2 fail. In the feedback channel, suppose that a NACK-to-ACK error occurs in process 1, and processes 2 and 3 successfully receive NACK for packets 1 and 2. The reordering buffer currently contains packet 3 together with the three gaps of packets 0 $\sim$ 2. At this moment, processes 1, 2, and 3 assume that the missing packets 0 $\sim$ 2 will be retransmitted, but are unsure which packet will be received. The stall avoidance scheme starts monitoring the statuses of processes 1, 2, and 3 to check whether these gaps belong to Type-II gaps.
2. In cycle 2, processes 1, 2, and 3 receive a new packet 4 and old packets 1 and 2, respectively. Now, the holes of packets 1 and 2 in the reordering buffer are filled. Since the NDI statuses of processes 1 and 4 are NEW and the TSNs of processes 2 and 3 indicate that packets 1 and 2 are received, the subsequent packets arriving at the receiver will have TSNs higher than 4. Thus, no process is responsible for sending the missing packet 0. Consequently, it can be confirmed that the gap of packet 0 is a Type-II gap. Hence, the available in-sequence packets 1 $\sim$ 4 are forwarded to the upper layer together with packet 0.

4 Performance Measure and System Assumptions

4.1 Gap Processing Time

To measure the performance of the stall avoidance schemes, we define a new performance measure—*gap processing time (GPT)*. It is the duration from the occurrence of a Type II gap until a stall avoidance scheme recognizes the existence of nonrecoverable gaps. The gap processing time is an important performance metric to evaluate the quality of service (QoS) for HSDPA. On the one hand, when the gap processing time is shorter than the RLC timeout, with the help of the stall avoidance scheme, an RLC retransmission can be initiated sooner when the Type-II gap is detected. This is especially important for the delay sensitive services. On the other hand, without the help of the stall avoidance scheme, the excessive gap processing time may be longer than the RLC timeout. When a packet is sent out, an RLC timer is started at the transmitter. If the acknowledgement is not received correctly before the RLC timer is expired, the transmitter sends the packet again. When a Type-II gap halts the received packets at the MAC layer to be delivered to the RLC layer and the excessive gap processing time is longer than the RLC timeout, the so called "spurious" RLC retransmission occurs. In this situation, the RLC timeout triggers the retransmission of some already received packets, which are halted at the MAC layer due to an unrecoverable Type-II gap. Therefore, the analysis of gap processing time offered in the paper can be helpful in setting an appropriate value of the RLC timeout from a viewpoint of the receiver's MAC layer performance.

Although the probability of the NACK-to-ACK error is 0.01 [24], [25], the high allowable packet error rate in the MAC layer of HSDPA can still lead to serious stall problems. To boost the transmission rate, the packet error rate of the first transmission can be higher than 40 percent because a high level modulation and coding scheme is adopted [22]. For example, when the probability of a NACK-to-ACK error is equal to 0.01, the average packet error rate is equal to 0.3, and at the transmission time interval of 2 milliseconds, the stall problem occurs 7.5 ($5/0.002 \times 0.3 \times 0.01$) times within 5 seconds. Thus, the impact of the NACK-to-ACK error cannot be ignored, especially for supporting delay sensitive services.

The stall problem affects the performance of the parallel HARQ mechanism in two folds. From the goodput aspect, the end-to-end goodput from the receiver's upper-layer viewpoint can be seriously degraded due to the long gap processing time. From the delay aspect, the end-to-end data delivery delay can also become longer due to the gap processing time. Referring to the analytical model in [33], the packet delivery delay (denoted by T_d) in the third-generation (3G) WCDMA with transport control protocol (TCP) can be decomposed as

$$T_d = Q_d + R_d + N_d, \quad (1)$$

where Q_d, R_d, and N_d are the queuing delay, reordering delay, and the wireline network delay, respectively. However, the packet delivery delay of (1) does not consider the effect of NACK-to-ACK errors. To incorporate the effect of NACK-to-ACK errors into packet delivery delay, T_d can be modified as

$$T_d = Q_d + R_d + N_d + GPT. \quad (2)$$

Note that the reordering delay R_d is caused by Type-I gaps while the GPT is caused by Type-II gaps. Clearly, a longer period of gap processing time causes longer packet delay because Type-II gaps halt the procedure of forwarding the received packets to the upper layer.

A longer period of gap processing time also results in more accumulated packets in the MAC layer. As a result, the overflow probability of the reordering buffer is increased. Moreover, as more received packets are forwarded to the RLC layer due to longer gap processing time, a larger buffer in the RLC layer is required to accommodate these packets. With large enough buffers of both MAC and RLC layers, the received packets can be accommodated in the receiver. In this paper, we focus on the analysis of the gap processing time for the three stall avoidance schemes. Gap-processing time is influenced by the physical layer parameters, such as packet error rate (PER) and the probability of a NACK becoming an ACK ($P_{N\to A}$), and the MAC layer parameters, e.g., the size of the reordering buffer and the number of processes in the parallel SAW HARQ mechanisms.

4.2 Assumptions

Being a function of both physical layer and MAC layer parameters, gap processing time is difficult to compute analytically. To make the analysis tractable, we make the following assumptions:

1. Because a NACK-to-ACK error usually occurs when a mobile terminal moves at high speeds, it is assumed that the fast changing channel is modeled by an independent Rayleigh fading channel from one packet to another packet.
2. All packets are assumed to have the same priority.
3. All transmit processes in the HARQ mechanism always have packets ready for transmission.
4. Effects of incremental redundancy and Chase combining are not considered. The provided analysis in the paper can be viewed as the worst-case analysis.
5. Assume the modulation and coding scheme and the packet length are not changed during the period of the gap processing time.
6. The feedback delay is not taken into account of the gap processing time.

5 Analysis of Timer-Based Stall Avoidance Scheme

In Proposition 1, we derive the average gap processing time for the timer-based stall avoidance scheme.

Proposition 1. *Denote P_s and $P_{N\to A}$ as the probability of a packet being successfully received and that of having a NACK-to-ACK error, respectively. Then, the probability with a Type-II gap (denoted by P_G) is equal to*

$$P_G = (1 - P_s)P_{N\to A}. \quad (3)$$

Let D be the expiry time of the timer normalized to the transmission time interval (TTI). In terms of P_G, P_s, and D, the average gap processing time for the timer-based stall avoidance scheme normalized to TTIs (denoted by $\overline{GPT}_{timer}$) in the single user case can be expressed as

$$\overline{GPT}_{timer} = \sum_{\ell=0}^{D} \overline{GPT}(\ell), \quad (4)$$

where the average gap processing time with ℓ Type-II gaps (denoted by $\overline{GPT}(\ell)$) is

$$\overline{GPT}(\ell) = \begin{cases} (1-P_G)^D \left[D + \sum_{i=1}^{\infty} i(1-P_s)^{i-1} P_s \right], & \ell = 0; \\ P_G^{\ell}(1-P_G)^{D-\ell} \sum_{t_1=1}^{D-\ell+1} \sum_{t_2=t_1+1}^{D-\ell+2} \cdots \sum_{t_\ell=t_{\ell-1}+1}^{D} \sum_{j=1}^{\ell} \frac{2D-t_j}{\ell}, & \ell = 1, \cdots, D, \end{cases} \quad (5)$$

and t_j is defined as the elapsed time of the previous timer until the end of gap j ($j = 1, \cdots, \ell$).

Proof. Assume that a timer is already initiated for a certain gap. We consider the following two scenarios:

1. *No new Type-II gap occurs in a period of D, $\ell = 0$.* Referring to Fig. 4a, let gap 0 occur after the previous timer expires. Suppose that i TTIs later, a packet PKT^* with a higher TSN than gap 0's is successfully received. Then, a new timer is initiated for gap 0. In this case, the gap processing time of gap 0 is i plus the expiration time of the timer D. Now, we approximate the probability of receiving PKT^* after i TTIs by $(1-P_s)^{i-1}P_s$. Note that the successfully received PKT^* may not always have a TSN higher than that of gap 0. Because the probability of not having new Type-II gap within D TTIs is $(1-P_G)^D$, the probability mass function (pmf) for the gap processing time (GPT) equal to $D+i$ TTIs can be expressed as

$$P(GPT = i + D) = (1-P_G)^D (1-P_s)^{i-1} P_s. \quad (6)$$

The average gap processing time in the case of $l = 0$ can be computed as follows:

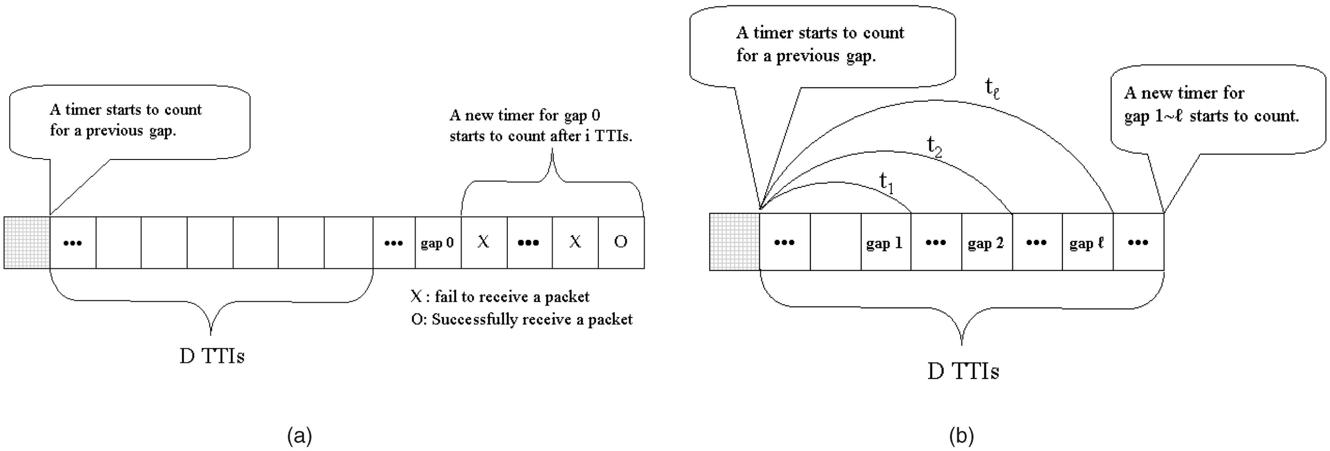

Fig. 4. Two scenarios for the timer-based stall avoidance scheme to remove a Type-II gap. (a) No new Type-II gap occurs before the time expires for a previous gap. (b) Some new Type-II gaps occur before the time expires for a previous gap.

$$\begin{aligned}\overline{GPT}(0) &= \sum_{i=1}^{\infty}(i+D)P(GPT=i+D)\\ &= \left[D+\sum_{i=1}^{\infty} i(1-P_s)^{i-1}P_s\right](1-P_G)^D.\end{aligned} \tag{7}$$

Thus, the first part of Proposition 1 is proven.

2. *ℓ Type-II gaps occur before the timer expires, $\ell \neq 0$.* Assume that ℓ new Type-II gaps occur before the timer expires, as shown in Fig. 4b. For gap j, ($j = 1, \ldots, \ell$), a period of time $(D - t_j)$ is needed before its own timer starts. Recall that t_j is defined as the elapsed time of the previous timer until the end of gap j. In this case, the total gap processing time for gap j is equal to $(2D - t_j)$ and the probability of having ℓ Type-II gaps conditioned on known $t_1, \cdots, t_\ell$ is $P_G^\ell (1-P_G)^{D-\ell}$. Note that the value of $P_G^\ell(1-P_G)^{D-\ell}$ is small for $\ell \geq 2$ in real cases. Thus, we can approximate the exact gap processing time for the ℓ gaps by an average value over the ℓ gaps as

$$\overline{GPT}(\ell|t_1,\cdots,t_\ell) = \sum_{j=1}^{\ell}\frac{2D-t_j}{\ell}. \tag{8}$$

Then, the pmf of the gap processing time for having ℓ gaps conditioned on known $t_1, \cdots, t_\ell$ can be expressed as

$$P(GPT = \overline{GPT}(\ell|t_1,\cdots,t_\ell)) = P_G^\ell(1-P_G)^{D-\ell}. \tag{9}$$

By considering all the possible occurrence time of ℓ Type-II gaps in D TTIs, we have

$$\begin{aligned}\overline{GPT}(\ell) = &\sum_{t_1=1}^{D-\ell+1}\sum_{t_2=t_1+1}^{D-\ell+2}\cdots\sum_{t_\ell=t_{\ell-1}+1}^{D}\sum_{j=1}^{\ell}\frac{2D-t_j}{\ell}\\ &\times P(GPT=\overline{GPT}(\ell|t_1,\cdots,t_\ell))\\ &= P_G^\ell(1-P_G)^{D-\ell}\sum_{t_1=1}^{D-\ell+1}\sum_{t_2=t_1+1}^{D-\ell+2}\cdots\sum_{t_\ell=t_{\ell-1}+1}^{D}\sum_{j=1}^{\ell}\frac{2D-t_j}{\ell}.\end{aligned} \tag{10}$$

Hence, we prove the second part of Proposition 1.□

6 Analysis of Window-Based Stall Avoidance Scheme

Proposition 2, we derive the average gap processing time for the window-based stall avoidance scheme.

Proposition 2. *Let W be the detection window size of the window-based stall avoidance scheme for an M-channel SAW HARQ mechanism. Denote P_{new} and P_{old} as the probability of receiving a new packet and that of receiving a retransmitted old packet, respectively. From the definitions of P_s and $P_{N\to A}$ in Proposition 1, P_{new} and P_{old} can be expressed as*

$$P_{new} = P_s + (1-P_s)P_{N\to A} \tag{11}$$

and

$$P_{old} = (1-P_s)(1-P_{N\to A}). \tag{12}$$

Then, the average gap processing time of the window-based stall avoidance scheme normalized to TTI in the single user case can be computed in terms of parameters P_{new}, P_{old}, W, and M as follows:

$$\overline{GPT}_{window} = M + \sum_{m=1}^{M}\left[(W-m)\sum_{n=1}^{\infty} nP_{new}P_{old}^{n-1}\right]\binom{M-1}{m-1}P_{new}^{m-1}P_{old}^{M-m}. \tag{13}$$

Proof. The gap processing time for the window-based stall avoidance scheme is derived in two steps. In the following, we will prove that

$$\overline{GPT}_{window} = \textit{cycle duration} + \textit{residual detection time}, \tag{14}$$

where the cycle duration is defined in Section 3.3 and the residual detection time is defined as the extra time to detect a Type II gap in addition to one cycle duration.

1. *Cycle duration*. The minimum gap processing time for the window-based stall avoidance scheme is equal to one cycle duration (M TTIs). Consider the smallest detection window $W = M$. The detection window size (or, equivalently, the

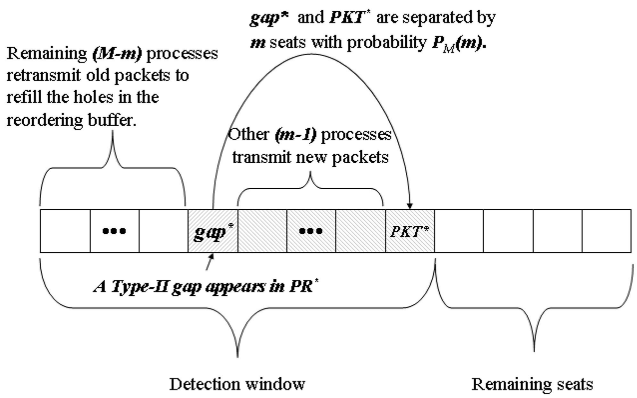

Fig. 5. An illustrative example of the seat allocation in a detection window for the window-based stall avoidance scheme.

reordering buffer size) is usually larger than the number of parallel HARQ processes, i.e., $W \geq M$. Suppose that a Type-II gap occurs in process PR^* in ith cycle. If subsequent $M-1$ processes transmit new packets successfully, process PR^* intends to send a new packet (PKT^*) in the $(i+1)$th cycle due to a NACK-to-ACK error. However, because the detection window is fully-booked, the gap in the ith cycle for process PR^* can be judged to be a Type-II gap according the rule of the window-based stall avoidance scheme. In this situation, the gap processing time is equal to one cycle duration (i.e., M TTIs).

2. *Residual detection time*. The *residual detection time* is defined as the extra time after one cycle duration for the detection window becoming fully-booked. The residual detection time spans in the following two cases: a. $W > M$ and b. $W = M$ with some of the $M-1$ processes subsequent to process PR^* receiving old packets after the occurrence of the Type-II gaps in step 1.

   a. $W > M$. Now, let us examine the seat allocation of a detection window, as shown in Fig. 5. Recall in step 1 that process PR^* produced a Type-II gap (gap^*) in cycle i. Thus, process PR^* will transmit a new packet PKT^* in cycle $i+1$. Without loss of generality, choose $(m-1)$ processes out of the total other $(M-1)$ processes to send new packets and the remaining $(M-m)$ processes to send old packets. Clearly, the probability of $(m-1)$ processes transmitting new packets is P_{new}^{m-1} and that of $(M-m)$ processes retransmitting old packets is P_{old}^{M-m}. Since there are $\binom{M-1}{m-1}$ choices, the probability of PKT^* and gap^* being separated by m seats is equal to

$$P_M(m) = \binom{M-1}{m-1} P_{new}^{m-1} P_{old}^{M-m}, \qquad (15)$$

where $\sum_{m=1}^{M} P_M(m) = 1$. A Type-II gap will finally halt the trailing edge of a detection window and is seated at the first place of the detection window. Hence, PKT^* will be positioned at the $(m+1)$th seat. Accordingly, there will be another $(W-m-1)$ available seats remained in the detection window. Note that the detection window extends its leading edge only when receiving a new packet and the old packets just fill in the empty holes in the reserved seats of the reordering buffer. Thus, the Type-II gap gap^* can be detected when the remaining $(W-m-1)$ plus one seats in the ordering buffer are occupied.

The gap processing time for gap^* will be $cycle\ duration + (W-m)$ TTIs when $W-m$ consecutive new packets arrive the receiver successfully with conditional probability P_{new}^{W-m}. Note that $cycle\ duration = M$. In this case, the probability of gap processing time equal to $M + (W-m)$ TTIs is $P_{new}^{W-m} P_M(m)$. However, the gap processing time will be longer than $M + (W-m)$ TTIs if one or more of the $W-m$ new packets fail the CRC at their first trials to reach the receiver. Thus, the gap processing time will be $M + (W-m) + 1$ TTIs if one of the $W-m$ new packets takes two transmissions (one transmission with probability P_{new} and one retransmission with probability P_{old}) to reach the receiver. In this case, the probability is $(W-m) P_{new}^{W-m} P_{old} P_M(m)$.

Now, we assume that there are $W-m$ transmissions and total $n-(W-m)$ retransmissions before these $W-m$ new packets successfully reach the receiver. The gap processing time is $n+M$ TTIs in this case. Because these total $n-(W-m)$ retransmissions may be responsible for some of the $W-m$ packets, there are $\frac{(n-1)!}{(W-m-1)!(n-(W-m))!} = \binom{n-1}{W-m-1}$ choices in this case, for each of which the probability is $P_{new}^{W-m} P_{old}^{n-(W-m)}$. Thus, the probability of the gap processing time equal to $n+M$ (denoted by $P(GPT = n+M)$) can be expressed as

$$\begin{aligned} &P(GPT = n+M) \\ &= P(GPT = n+M \mid PKT^* \text{ at } (m+1)\text{th seat}) P_M(m) \\ &= \binom{n-1}{W-m-1} P_{new}^{W-m} P_{old}^{n-(W-m)} \binom{M-1}{m-1} P_{new}^{m-1} P_{old}^{M-m} \qquad (16) \\ &= \binom{n-1}{W-m-1} \binom{M-1}{m-1} P_{new}^{W-1} P_{old}^{n+M-W}. \end{aligned}$$

Note that

$$\sum_{m=1}^{M} \sum_{n=W-m}^{L} P(GPT = n+M) = 1, \qquad (17)$$

where L is the required TTIs to detect a Type-II gap. The average gap processing time for the case of $W > M$ can be calculated by

$$\begin{aligned}&\overline{GPT}_{window}\\&=\sum_{m=1}^{M}\sum_{n=W-m}^{L}(n+M)P(GPT=n+M)\\&=\sum_{m=1}^{M}\sum_{n=W-m}^{L}(n+M)\binom{n-1}{W-m-1}\binom{M-1}{m-1}P_{new}^{W-1}P_{old}^{n+M-W}.\end{aligned}\tag{18}$$

b. W = M. Recall that the residual time for $W = M$ lasts when some of the $M-1$ processes subsequent to process PR^* receive old packets after the occurrence of the Type-II gaps in step 1. Following the same approach of obtaining (15) in the case of $W > M$, we assume that the $(m-1)$ processes of the subsequent $(M-1)$ processes send new packets and the remaining $(M-m)$ processes send old packets. In this situation, gap^* can be recognized as a Type-II gap when the remaining $(M-m-1)$ plus one seats in the ordering buffer are occupied. Then, one can easily find that the residual time for $W = M$ is just a special case of that for $W > M$. However, one should note that (16) is not valid for $W = M$ and $m = M$. For $W = M$ and $m = M$, $P(GPT = n + M|PKT^* \text{ at } (M+1)\text{th seat}) = P_{new}P_{old}^{n}$ because PKT^* may successfully reach the receiver and overbook the detection window after $n+1$ transmissions, where $n \geq 0$.

In order to obtain a general closed-form expression of average gap processing time for the window-based stall avoidance scheme, an alternative derivation of Proposition II is provided in the following. Since the probability of a new packet arriving at the reordering buffer within n TTIs is equal to $P_{new}P_{old}^{n-1}$, the average time for the detection window extending its leading edge by one seat (denoted by T_0) can be computed by

$$T_o=\sum_{n=1}^{\infty}nP_{new}P_{old}^{n-1}.\tag{19}$$

As a result, it takes extra $(W-m-1)T_o$ TTIs to have a fully-booked window in addition to one cycle duration. When another new packet arrives in the next T_o, the receiver can judge the gap at the first seat of the window is a Type-II gap and start the process of forwarding the received packets with the Type-II gap to the upper later. From (15) and (19) and the above discussion, the total residual detection time in addition to one cycle duration is calculated as

$$\begin{aligned}&residual\,detection\,time=\sum_{m=1}^{M}[(W-m)T_o]P_M(m)\\&=\sum_{m=1}^{M}\left[(W-m)\sum_{n=1}^{\infty}nP_{new}P_{old}^{n-1}\right]\binom{M-1}{m-1}P_{new}^{m-1}P_{old}^{M-m}.\end{aligned}\tag{20}$$

Adding the residual detection time (20) to the cycle duration (M TTIs), we obtain the closed-form expression of the average gap processing time for the window-based stall avoidance scheme as shown in (13). Note that the average gap processing time calculated by (20) plus the cycle duration M is equal to that obtained in (18). □

7 Analysis of Indicator-Based Stall Avoidance Scheme

In Proposition 3, we derive the average gap processing time for the indicator-based stall avoidance scheme.

Proposition 3. *Consider an M-channel SAW HARQ mechanism. The gap processing time normalized to TTIs for the indicator-based scheme in the single user case can be calculated in terms of parameters M, P_s, P_{new}, and P_{old} as follows:*

$$\begin{aligned}\overline{GPT}_{indicator}&=M(P_{new}+P_{old}P_s)^{M-1}\\&+\sum_{k=1}^{C}\sum_{m=2}^{M}(m-1+kM)P(x_k=S_1)\\&\left[\sum_{i=0}^{k}P(x_i=S_1)\right]^{m-2}\left[\sum_{j=0}^{k-1}P(x_j=S_1)\right]^{M-m},\end{aligned}\tag{21}$$

where

$$P(x_0=S_1)=P_{new}+P_{old}P_s\tag{22}$$

and

$$P(x_\ell=S_1)=P_{old}^{\ell}(1-P_s)P_{new}\quad for\quad \ell\geq 1.\tag{23}$$

C is the number of cycles involving all the processes in the M-channel SAW HARQ mechanism required to detect a Type-II gap.

Proof. Recall that the basic idea of the indicator-based stall avoidance scheme is to examine the status of each HARQ process by the use of the NDI in the control channel and the TSN in the traffic channel to confirm whether the missing packet will be transmitted or not. To ease our discussion, the status of an HARQ process is described by the following two events:

$$\mathbf{A}\triangleq\{NDI_{(rec)}=NEW\};\tag{24}$$

$$\mathbf{B}\triangleq\{\{TSN_{(rec)}\neq TSN^*\}\cap\{NDI_{(rec)}=OLD\}\},\tag{25}$$

where $NDI_{(rec)}$ and $TSN_{(rec)}$ are the NDI and the TSN of the received packet; TSN^* is the TSN of the missing gap. If either event $\mathbf{A}$ or event $\mathbf{B}$ is sustained for a HARQ process, then this HARQ process is ruled out to be the candidate for sending the missing packet TSN^*. If all the HARQ processes are ruled out, it implies that the existing gap in the reordering buffer belongs to the nonrecoverable Type-II gap since no HARQ process will transmit the missing packet. However, if none of events $\mathbf{A}$ and $\mathbf{B}$ happens (denoted by $\overline{\{\mathbf{A}\cup\mathbf{B}\}}$), the receiver believes that the HARQ process still possibly transmits the packet TSN^* to fill the gap of the reordering buffer in

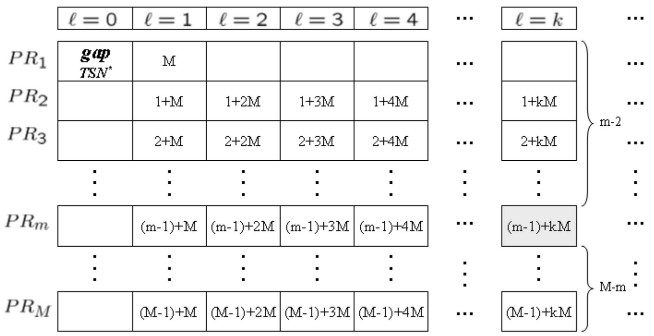

Fig. 6. An illustration for the gap processing time of the indicator-based stall avoidance scheme.

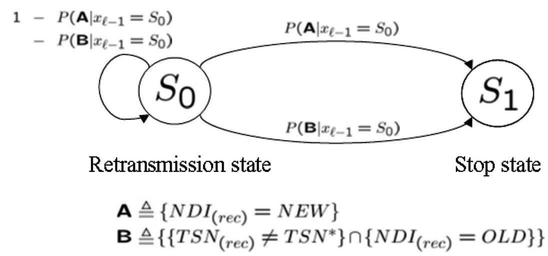

Fig. 7. The state transition diagram for the indicator-based stall avoidance scheme.

the future. Note that when $TSN_{(rec)} = TSN^*$, this Type-I gap is filled in the reordering buffer, which will not cause the stall issue.

Assume that a Type-II gap appears in the first process (PR_1) of the M-processes SAW HARQ mechanism in the cycle 0 ($\ell = 0$) as shown in Fig. 6. To begin the proof, we first define a semi-Markov chain to describe the status of each HARQ process. Fig. 7 shows this Markov chain with two states defined as follows:

1. ***Retransmission state*** (S_0). If event $\overline{\{\mathbf{A} \cup \mathbf{B}\}}$ happens, this HARQ process enters the retransmission state. Denote x_ℓ as the state variable in the lth cycle. If a process is in the retransmission state in the $(\ell-1)$th cycle, the receiving process will issue a NACK signal for requesting a retransmission in the ℓth cycle. However, if the NACK signal is changed to an ACK signal, a new packet will be transmitted in the ℓth cycle instead. Based on the definition of event **A** in (24), we have

$$P(\mathbf{A}|x_{\ell-1} = S_0) = P_{N\to A}, \tag{26}$$

where $P_{N\to A}$ is the probability of having a NACK-to-ACK error. On the contrary, if the NACK signal arrives the transmitter correctly, the missing packet will be retransmitted in the ℓth cycle. Note that the probability of the NACK signal arriving the transmitter correctly is $(1 - P_{N\to A})$ and the probability of successfully retransmitting packet to the receiver is P_s. According to the definition of (25), the occurrence probability of event **B** is equal to

$$P(\mathbf{B}|x_{\ell-1} = S_0) = (1 - P_{N\to A})P_s. \tag{27}$$

Because events **A** and **B** are mutually exclusive, from (26) and (27), we can obtain

$$\begin{aligned} P(x_\ell = S_0|x_{\ell-1} = S_0) &= P(\overline{\{\mathbf{A} \cup \mathbf{B}\}}|x_{\ell-1} = S_0) \\ &= 1 - [P(\mathbf{A}|x_{\ell-1} = S_0) + P(\mathbf{B}|x_{\ell-1} = S_0)] \\ &= 1 - [P_{N\to A} + (1 - P_{N\to A})P_s] \\ &= (1 - P_{N\to A})(1 - P_s) \\ &= P_{old}. \end{aligned} \tag{28}$$

Recall that, in (12), we define

$$P_{old} = (1 - P_{N\to A})(1 - P_s).$$

According to the definition of S_0, when a process receives a retransmitted old packet which fails the CRC, it enters state S_0. Thus, the initial probability of a process at S_0 can be expressed as

$$P(x_0 = S_0) = P_{old}(1 - P_s). \tag{29}$$

2. ***Stop state*** (S_1). If either event **A** or event **B** happens, the state of process will enter the stop state. From Fig. 7, (26), and (27), the probability of a process in the stop state S_1 in the ℓth cycle can be expressed as

$$\begin{aligned} P(x_\ell = S_1|x_{\ell-1} = S_0) &= P(\mathbf{A} \cup \mathbf{B}|x_{\ell-1} = S_0) \\ &= P(\mathbf{A}|x_{\ell-1} = S_0) + P(\mathbf{B}|x_{\ell-1} = S_0) \\ &= P_{N\to A} + (1 - P_{N\to A})P_s \\ &= P_s + (1 - P_s)P_{N\to A} \\ &= P_{new}. \end{aligned} \tag{30}$$

Recall that, in (11), we define

$$P_{new} = P_s + (1 - P_s)P_{N\to A}.$$

According to the definition of state S_1, when a process receives a new packet or successfully receives a retransmitted old packet, it will enter state S_1. Thus, the initial probability of a process at S_1 can be expressed as

$$P(x_0 = S_1) = P_{new} + P_{old}P_s. \tag{31}$$

From (28) and (30), we obtain the state transition probability matrix $\mathbf{\Gamma}$ of the two-state Markov chain in Fig. 7 as follows:

$$\begin{aligned} \mathbf{\Gamma} &\triangleq \begin{bmatrix} P(x_\ell = S_0|x_{\ell-1} = S_0) & P(x_\ell = S_1|x_{\ell-1} = S_0) \\ P(x_\ell = S_0|x_{\ell-1} = S_1) & P(x_\ell = S_1|x_{\ell-1} = S_1) \end{bmatrix} \\ &= \begin{bmatrix} P_{old} & P_{new} \\ 0 & 0 \end{bmatrix}. \end{aligned} \tag{32}$$

Then, from (29) and (31), a state probability vector $\mathbf{P}(\mathbf{x}_\ell) \triangleq [P(x_\ell = S_0),\ P(x_\ell = S_1)]$ can be obtained by

$$\begin{aligned} \mathbf{P}(\mathbf{x}_\ell) &= \mathbf{P}(\mathbf{x}_{\ell-1})\mathbf{\Gamma} = \mathbf{P}(\mathbf{x}_0)\mathbf{\Gamma}^\ell \\ &= [P_{old}^{\ell+1}(1 - P_s),\ P_{old}^{\ell}(1 - P_s)P_{new}]. \end{aligned} \tag{33}$$

Now, referring to Fig. 6, we derive the average gap processing time for the following two possible scenarios:

1. ***Detect Type-II gap right after cycle 0.*** If, in cycle 0, processes $PR_2 \sim PR_M$ are all in state S_1, the gap TSN^* can be detected as a Type-II gap right after the end of cycle 0. Due to a NACK-to-ACK error, PR_1 transmits a new packet in cycle 1. As long as process PR_1 receives the NDI and finds $NDI = NEW$, the receiver can confirm that the missing packet TSN^* is a Type-II gap since all processes are handling other packets except for TSN^*. In this case, the receiver spends M TTIs to detect this Type-II gap. Thus, from (31), the probability of gap processing time equal to M can be expressed as

$$\begin{aligned} P(GPT = M) &= [P(x_0 = S_1)]^{M-1} \\ &= (P_{new} + P_{old}P_s)^{M-1}. \end{aligned} \tag{34}$$

Then, the average gap processing time for detecting a Type-II gap immediately after cycle 0 can be calculated as

$$\begin{aligned} \overline{GPT}_0 &= M \times P(GPT = M) \\ &= M(P_{new} + P_{old}P_s)^{M-1}. \end{aligned} \tag{35}$$

2. ***Detect the Type-II gap more than one cycle.*** Recall that if a process is in state S_1, this process is ruled out to be the candidate for sending the missing packet TSN^*. When all the processes are all ruled out, the remaining gap in the reordering buffer is the Type-IIgap. As shown in Fig. 6, $GPT = m - 1 + kM$ when the Type-II gap is detected in the kth cycle by PR_m, where $m \geq 2$ and $k \geq 1$. To compute the probability of $GPT = m - 1 + kM$, three events, E_1, E_2, and E_3, are defined as follows:

$$E_1 = \{PR_m \text{ is ruled out in the } k\text{th cycle.}\} \tag{36}$$

$$E_2 = \{PR_2 \sim PR_{m-1} \text{ are ruled out within } k\text{th cycle.}\} \tag{37}$$

$$E_3 = \{PR_{m+1} \sim PR_M \text{ are ruled out within} (k-1)\text{th cycle.}\} \tag{38}$$

From the state probability of the Markov chain in (33), it is obvious that

$$P(E_1) = P(x_k = S_1). \tag{39}$$

Before calculating $P(E_2)$, we first denote $P_k^{(\alpha)}$ the probability of process PR_α being ruled out within kth cycle. Because all processes send data independently, the probability for processes $PR_2 \sim PR_{m-1}$ being ruled out within kth cycle can be expressed as

$$\begin{aligned} P(E_2) &= \overbrace{P_k^{(2)} \times P_k^{(3)} \times \cdots \times P_k^{(m-1)}}^{(m-2)} \\ &= \left[\sum_{i=0}^{k} P(x_i = S_1)\right]^{m-2}. \end{aligned} \tag{40}$$

Similarly, for event E_3, we can have

$$P(E_3) = \left[\sum_{j=0}^{k-1} P(x_j = S_1)\right]^{M-m}. \tag{41}$$

Combining (39), (40), and (41), we can obtain the pmf of the GPT of the indicator-based stall avoidance scheme as

$$\begin{aligned} &P(GPT = m - 1 + kM) = \\ &P(x_k = S_1)\left[\sum_{i=0}^{k} P(x_i = S_1)\right]^{m-2}\left[\sum_{j=0}^{k-1} P(x_j = S_1)\right]^{M-m}. \end{aligned} \tag{42}$$

Then, we can express the average gap processing time for case 2 as follows:

$$\begin{aligned} \overline{GPT}_1 &= \sum_{k=1}^{C}\sum_{m=2}^{M}(m-1+kM)P(GPT = m-1+kM) \\ &= \sum_{k=1}^{C}\sum_{m=2}^{M}(m-1+kM)P(x_k = S_1) \\ &\quad \left[\sum_{m=2}^{k} P(x_i = S_1)\right]^{m-2}\left[\sum_{j=0}^{k-1} P(x_j = S_1)\right]^{M-m}, \end{aligned} \tag{43}$$

where C is the number of cycles involving all the processes in the M-channel SAW HARQ mechanism required to remove a Type-II gap. For given parameters P_{new}, P_{old}, P_s, and M, the value of C can be obtained from

$$\begin{aligned} &(P_{new} + P_{old}P_s)^{M-1} + \sum_{k=1}^{C}\sum_{m=2}^{M} P(x_k = S_1) \\ &\left[\sum_{i=0}^{k} P(x_i = S_1)\right]^{m-2}\left[\sum_{j=0}^{k-1} P(x_j = S_1)\right]^{M-m} = 1. \end{aligned} \tag{44}$$

Combining (35) and (43), the average gap processing time for the indicator-based stall avoidance scheme can be computed by

$$\begin{aligned} \overline{GPT}_{indicator} &= \overline{GPT}_0 + \overline{GPT}_1 \\ &= M(P_{new} + P_{old}P_s)^{M-1} \\ &\quad + \sum_{k=1}^{C}\sum_{m=2}^{M}(m-1+kM)P(x_k = S_1) \\ &\quad \left[\sum_{i=0}^{k} P(x_i = S_1)\right]^{m-2}\left[\sum_{j=0}^{k-1} P(x_j = S_1)\right]^{M-m}. \end{aligned} \tag{45}$$

□

TABLE 2
The Simulation Environment

Channel model	Rayleigh fading
Doppler frequency	100 Hz
Packet size	320 bits
CRC bits	16 bits
No. of parallel HARQ processes	4, 6, 8
TTI	2 msec
E_b/N_0 (dB)	$10 \sim 20$ dB
1^{st} transmission PER [1]	$0.09 \sim 0.42$

[1]To boost the transmission rate, the packet error rate of the first transmission can be higher than 40 percent because a high level modulation and coding scheme is adopted [22].

8 Numerical Results and Discussions

In this section, by analysis and simulations, we investigate the average gap processing time and the corresponding probability mass functions for the timer-based, the window-based, and the indicator-based stall avoidance schemes. We will discuss the relation between the gap processing time, the allowable retransmissions, and the number of acceptable fully loaded users for different stall avoidance schemes. For these purposes, we conduct a simulation cross physical and MAC layers under the assumptions of Section 4.2. In the simulation, each packet is transmitted through the flat Rayleigh fading channel. The Doppler frequency of the Rayleigh fading channel is 100 Hz, which is equivalent to a vehicle speed of 54 km/hour with a carrier frequency of 2 GHz. The correctness of each received packet is checked by CRC. If the received packet passes CRC, an ACK signal will be sent through the feedback control channel otherwise a NACK signal is sent. If a NACK-to-ACK error occurs in the feedback control channel, a Type-II gap will appear in the reordering buffer. The gap processing time of the Type-II gap is calculated from its appearance until it is detected. To confirm the correctness of simulation results, we run 100,000 packets for each simulation. The simulation parameters are shown in Table 2. In the discussion of the number of acceptable fully loaded users, we assume that all fully loaded users are selected from multiple users and under the same QoS requirements, i.e., the required gap processing time should be lower than 100 TTIs. A fair scheduling policy is implemented to allocate resource to fully loaded users.

8.1 Average Gap Processing Time of the Timer-Based Scheme

Fig. 8 shows the average gap processing time of the timer-based stall avoidance scheme with various settings on the timer's expiration (D). It is shown that the analytical results are close to the simulation results. In the case of $D = 20$ TTIs and $E_b/N_0 = 14$ dB, the analytical average gap processing time (21.7 TTIs) is only 2.7 percent smaller than the simulation value (22.3 TTIs). Regarding the 2.7 percent

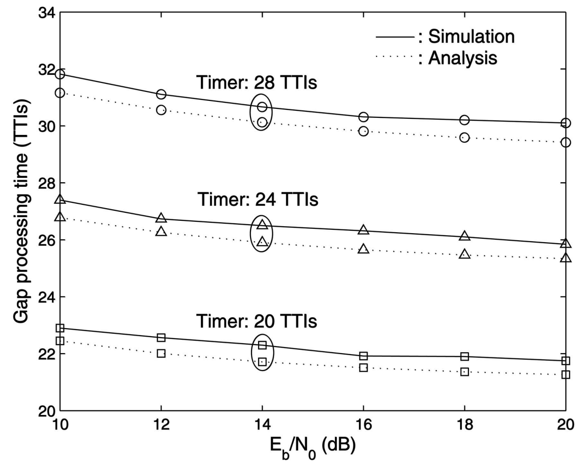

Fig. 8. The average gap processing time of the timer-based stall avoidance scheme with different timer expiration for the 4-channel SAW HARQ mechanism in the Rayleigh fading channel with Doppler frequency of 100 Hz.

discrepancy between the simulation and the analytical results, we summarized the reason as what follows.

To ease the derivation of the computational formula of the gap processing time for the timer-based avoidance schemes, we assume that the transmission sequence number (TSN) of the missing packet is smaller than that of the first successfully received consequent packet. Having a packet with a larger TSN after the missing packet with a smaller TSN, the receiver can find a gap in a series of received packets, thereby initiating the timer counting for the timer-based scheme. In general, it is possible that the receiver may receive a packet with a smaller TSN than the previous missing packet. In this case, the receiver does not know to initiate the timer counting for the missing gap until the reception of another packet with a larger TSN than the previous missing packet's TSN. However, the analysis of gap processing time for the general case is quite involved, which is not discussed in this paper. Hence, Proposition 1 shows the lower bound on the average gap processing time for the timer-based stall avoidance mechanisms.

The timer's expiration impacts the MAC layer performance of HSDPA in two folds. On the one hand, as shown in the figure, a longer timer results in longer gap processing time. For $D = 20$ TTIs at $E_b/N_0 = 14$ dB, the average gap processing time ($\overline{GPT}_{timer}$) is 21.7 TTIs, while for $D = 28$ TTIs, the $\overline{GPT}_{timer} = 30.1$ TTIs. On the other hand, a larger value of time expiration allows more retransmissions. Specifically, for an M-process SAW HARQ mechanism, the allowable number of retransmissions (h) is equal to D/M in the single user case. A properly designed timer expiration is to allow enough retransmissions to recover the lost packet and, in the meanwhile, not to cause too long of a gap processing time. From Proposition 1, we can rapidly evaluate the impact of the allowable retransmissions $(h = D/M)$ on the gap processing time ($\overline{GPT}_{timer}$) for different values of P_s and $P_{N\to A}$ without time consuming simulations.

Furthermore, the derived analytical method of computing the gap processing time can also be applied to design

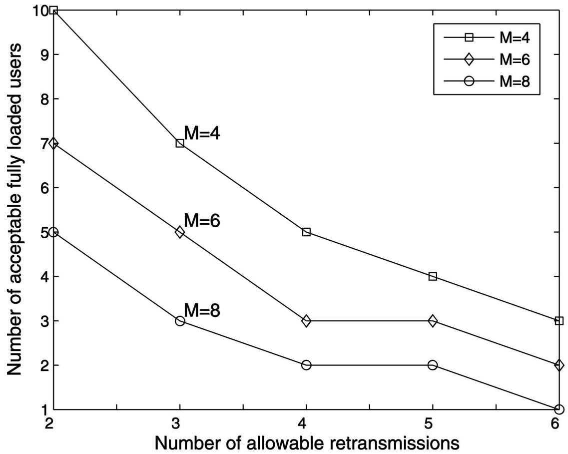

Fig. 9. The number of acceptable fully loaded users of the timer-based stall avoidance scheme versus the numbers of allowable retransmissions (h) for various number processes (M) in the parallel SAW HARQ mechanism subject to a constraint of gap processing time 100 TTIs.

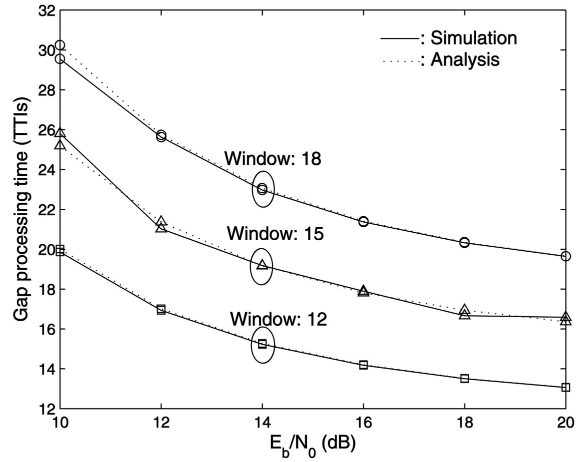

Fig. 10. The average gap processing time of the window-based stall avoidance scheme with different window sizes for the 4-channel SAW HARQ mechanism n the Rayleigh fading channel with Doppler frequency of 100 Hz.

the proper number of acceptable fully loaded users from the admission control standpoint. Let T_o be the constraint on the maximum allowable gap processing time. Then, the acceptable fully loaded users in the system can be approximated by $\lfloor T_o/\overline{GPT}_{timer} \rfloor$, where $\lfloor x \rfloor$ is the largest integer less than or equal to x. According to the above guidelines, Fig. 9 shows the number of acceptable fully loaded users of the timer-based stall avoidance scheme versus the number of allowable retransmissions (n) for different numbers of parallel processes (M) subject to the constraint of $T_o = 100$ TTIs. As shown in the figure, the more the allowable retransmissions (n), the fewer the users can be accepted. For $M = 4$, as the value of h increases from 2 to 6, the number of acceptable fully loaded users decreases from 10 to 3. This is because, for a given M, a larger value of h leads to a longer timer expiration ($D = n \times M$ in the single user case), thereby resulting in a longer period of the gap processing time. That is, more retransmissions increases the gap processing time and, thus, decreases the number of acceptable fully loaded users. Similarly, as the number of parallel HARQ processes increases, the effect of increasing gap processing time will reduce the acceptable fully loaded users.

8.2 Average Gap Processing Time of the Window-Based Scheme

Fig. 10 shows $\overline{GPT}_{window}$ against E_b/N_0 for various window sizes. The accuracy of the derived average gap processing time (i.e., (13)) of the window-based stall avoidance scheme is validated by simulations. As shown in the figure, the analytical results match the simulations well. Most importantly, Fig. 10 also provides important insights into designing an admission control policy subject to the gap processing time constraint T_0. Consider a fair scheduling policy. The number of acceptable fully loaded users (N) in the system can be approximated by

$$N = \lfloor T_o/\overline{GPT}_{window} \rfloor, \tag{46}$$

where $\lfloor x \rfloor$ is the largest integer less than or equal to x. For $E_b/N_0 = 14$ dB in Fig. 10, one can observe that $\overline{GPT}_{window} = 15.3, 19.2, 23.1$ TTIs for $W = 12, 15, 18$, respectively. For the maximal gap processing time constraint $T_0 = 100$ TTIs, the allowable users are therefore equal to 6, 5, and 4 for the window of size of 12, 15, and 18, respectively.

As a matter of fact, the window size is a function of the allowable minimum retransmissions (n) and the number of HARQ processes (M). In the single user case, for an M-process SAW HARQ mechanism with a window of size of W, the allowable retransmissions (n) of a missing packet is at least

$$n = \frac{W}{M-1} - 1. \tag{47}$$

For $M = 4$ and $W = 12$, the missing packet can be retransmitted by $\frac{W}{M-1} - 1 = 3$ times. Assume one process keeps transmitting a missing packet and the other three processes transmit new packets successfully. After four cycles, a window with a size of 12 is fully occupied. Because the window has no more space for the missing packet, this packet will not be transmitted.

Hence, the admission control policy subject to the gap processing time requirement can also be designed for different combinations of parameters n and M. Take $M = 4$ as an example. Fig. 11 shows the number of acceptable fully loaded users versus E_b/N_0 with various minimum allowable retransmissions (n) subject to a gap processing time constraint of 100 TTIs. These curves are obtained by mapping Fig. 10 according to (46). From (47), $n = 3,\ 4,$ and 5 correspond to $W = 12,\ 15,$ and 18, respectively. As shown in the figure, the acceptable fully loaded users are reduced from 7 to 4 as n increases from 3 to 5 for $16\text{dB} \leq E_b/N_0 \leq 18\text{dB}$.

Fig. 12 shows the number of acceptable fully loaded users of the window-based stall avoidance scheme against packet error rate (PER) with various numbers of parallel processes (M), where the maximal allowable gap processing time $T_o = 100$ TTIs and the minimum allowable retransmission $n = 3$.

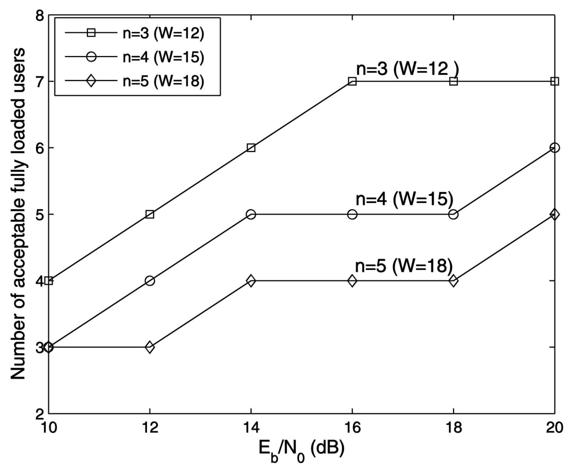

Fig. 11. The number of acceptable fully loaded users of the window-based stall avoidance scheme versus E_b/N_0 with various number minimum allowable retransmissions $(n = 3, 4, 5)$ in the 4-process SAW HARQ mechanism subject to a gap processing time constraint of 100 TTIs.

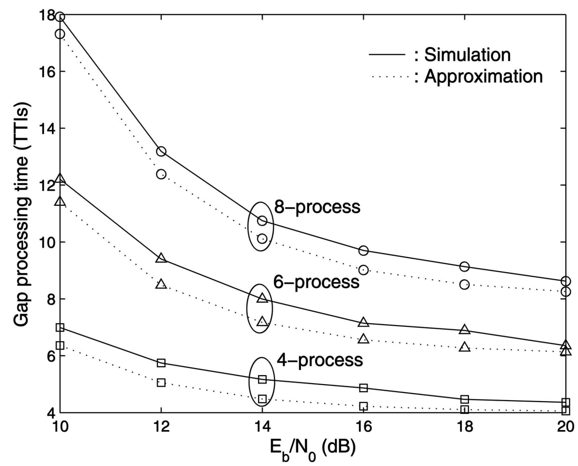

Fig. 13. Effect of the number of processes in the multichannel SAW HARQ mechanism on the gap processing time for the indicator-based avoidance scheme in the Rayleigh fading channel with Doppler frequency of 100 Hz.

These curves are obtained by substituting the parameters M, W, P_s, and $P_{N\rightarrow A}$ into (13). Note that $PER = 1 - Ps$ and the corresponding window size $W = 12, 20,$ and 28 is obtained from $W = (n + 1)(M - 1)$ according to (47). This figure can be associated with an admission control policy subject to gap processing time by observing PER. According to the CRC results and PER, a suitable number of allowable users to maintain the QoS can be determined from the standpoint of meeting the gap processing time requirement. In the figure, we find that more parallel SAW HARQ processes results in fewer allowable users in the system subject to the total gap processing time requirement. For $PER = 0.15$, the number of acceptable fully loaded users decreases from 7 to 3 as M increases from 4 to 8. Recall that $n = \frac{W}{M-1} - 1$ in (47). For a fixed n, a larger value of M also leads to a lager value of W and longer gap processing time. Hence, the number of acceptable fully loaded users is reduced for a larger value of M to satisfy the gap processing time requirement.

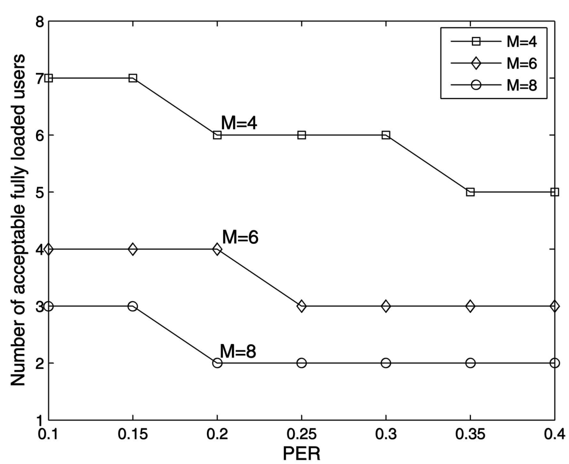

Fig. 12. The number of acceptable fully loaded users of the window-based stall avoidance scheme versus PER with various number processes (M) in the parallel SAW HARQ mechanism subject to a gap processing time constraint of 100 TTIs. The minimum allowable retransmission (n) is three.

8.3 Average Gap Processing Time of the Indicator-Based Scheme

Fig. 13 shows the analytical average gap processing time for the indicator-based stall avoidance scheme obtained from (21) and simulations. As shown in the figure, the differences between simulation and the analytical approximation are quite small. Similar to the reason for the timer-based scheme, the receiver may possibly receive a packet with a smaller TSN than the previous missing packet. In this situation, the procedure of monitoring the status of the HARQ process can start only after another packet with a larger TSN arrives. Hence, Proposition 3 shows a lower bound on the average gap processing time for the indicator-based stall avoidance mechanism.

Compared to Figs. 8, 10, and 13, the indicator-based scheme outperforms the timer-based and the window-based schemes in terms of the gap processing time. For a 4-process SAW HARQ mechanism with $E_b/N_0 = 14$ dB, the gap processing time of the indicator-based scheme is 4.9 TTIs; the gap processing time is 21.7 TTIs for the timer-based scheme with a timer expiration of 20 TTIs; the gap processing time is 15.27 TTIs for the window-based scheme with a window of a size of 12. Also, it is found that the more the parallel HARQ processes, the longer the average gap processing time.

The developed gap processing time computation method for the indicator-based scheme can be applied to determine the acceptable fully loaded users through $\lfloor T_o/\overline{GPT}_{indicator} \rfloor$ as the timer-based and the window-based schemes, where T_o is a given constrain on the gap processing time. Fig. 14 shows the number of acceptable fully loaded users of the indicator-based stall avoidance scheme against PER with various numbers of parallel HARQ processes (M) under a constraint of the gap processing time 100 TTIs. One can find that with the aid of the indicator-based stall avoidance

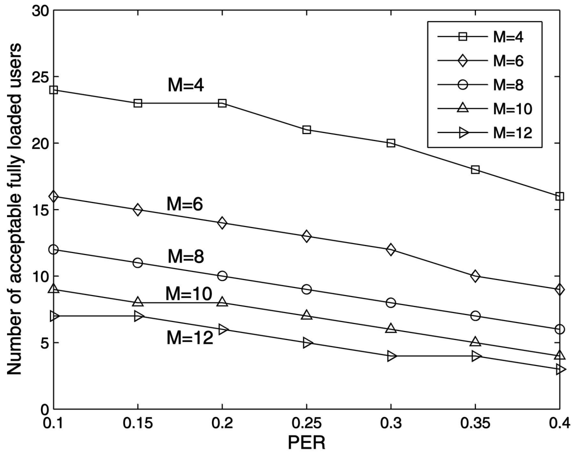

Fig. 14. The number of acceptable fully loaded users of the indicator-based stall avoidance scheme versus PER with various number processes (M) in the parallel SAW HARQ mechanism subject to a gap processing time constraint of 100 TTIs.

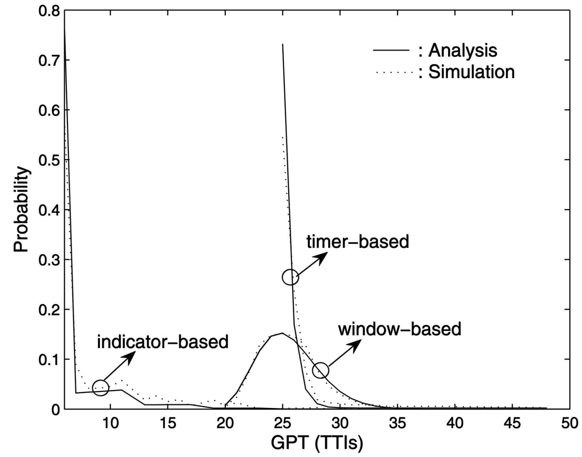

Fig. 15. The probability mass functions of the gap processing time for the timer-based, the window-based, and the indicator-based stall avoidance schemes, where the timer's expiration $D = 24$ and the detection window size $W = 20$ with $M = 6$ parallel HARQ processes at $E_b/N_0 = 14$ dB.

scheme, an HSDPA system can accommodate more users compared to the window-based scheme. For $M = 4$ at $PER = 0.2$, the number of acceptable fully loaded users are 4, 6, and 24 for the timer-based, the window-based, and the indicator-based schemes, respectively. Furthermore, although more parallel HARQ processes can enhance throughput, the side effect of increasing gap processing time can not be ignored. As M increases from 4 to 12 at $PER = 0.2$, it is necessary to reduce the acceptable fully loaded users from 24 to 6 if the gap processing time requirement is fulfilled.

8.4 Probability Mass Function of the Gap Processing Time

Fig. 15 shows the probability mass functions of the gap processing time for the timer-based, the window-based, and the indicator-based stall avoidance schemes, where the timer's expiration $D = 24$ and the detection window size $W = 20$ with $M = 6$ parallel HARQ processes at $E_b/N_0 = 14$ dB. The analytical pmf for the timer-based scheme can be obtained by evaluating (6) and (9). For the window-based scheme, we obtain the analytical pmf by evaluating (16), while the pmf for the indicator-based scheme can be obtained by calculating (35) and (43). From the figure, one can see that the analytical results can approximate the simulation results. Most importantly, the gap processing time for the timer-based and the indicator-based schemes are centralized, while that of the window-based scheme is widely spread. For example, 78 percent and 71 percent of the gap processing time of the timer-based and the indicator-based schemes are lower than 26 and 8 TTIs, respectively, where the corresponding average gap-processing times are 26.8 and 8.4 TTIs. However, for the window-based scheme, only 53 percent of the gap processing time are lower than the average value of 25.6 TTIs. Thus, we can conclude that indicator-based scheme is more capable of maintaining stable and better QoS for HSDPA.

9 Conclusions

In this paper, we have defined a new performance metric-gap processing time-to evaluate the stall avoidance schemes for HSDPA. We derive the probability mass functions and the closed-form expressions for the average gap processing time of the timer-based, the window-based, and the indicator-based stall avoidance schemes. Through analyses or simulations, we find that the indicator-based stall avoidance scheme outperforms the other two schemes in terms of the gap processing time. The proposed analytic model can also be used to design the proper size in the reordering buffer in the MAC layer, the window size in the RLC layers, the RLC timeout period, and the number of acceptable fully loaded users in the radio resource management layer. Some interesting future research topics that can be extended from this work include the joint design of the MAC and RLC retransmission mechanism and the investigation of the effect of Chase combining on the gap processing time and its related upper protocol layers.

Acknowledgments

The authors wish to thank Dr. Sam Jiang for his helpful suggestions and ASUSTek Computer Inc. for its sponsorship in this project. This work was supported jointly by the National Science Council ROC under contracts 93-2219-E-009-012, 92-2219-E-009-026, and EX-91-E-FA06-4-4, and the ASUSTek Computer Inc. under the contract 91C189. Part of this work appeared in the *Proceedings of the IEEE International Conference on Wireless Networks, Communications, and Mobile Computing*, vol. 1, pp. 81-86, June 2005.

References

[1] 3GPP TR 25.950 V4.0.0, *"UTRA High Speed Downlink Packet Access,"* Mar. 2001.
[2] S. Parkvall, E. Dahlman, P. Frenger, P. Beming, and M. Persson, "The High Speed Packet Data Evolution of WCDMA," *Proc. IEEE Int'l Symp. Personal, Indoor, and Mobile Radio Comm.*, pp. G27-G31, Sept. 2001.

[3] R.C. Qiu, W. Zhu, and Y.-Q. Zhang, "Third-Generation and Beyond (3.5G) Wireless Networks and Its Applications," *Proc. IEEE Int'l Symp. Circuits and Systems,* pp. I41-I44, May 2002.
[4] M. Döttling, J. Michel, and B. Raaf, "Hybrid ARQ and Adaptive Modulation and Coding Schemes for High Speed Downlink Packet Access," *Proc. IEEE Int'l Symp. Personal, Indoor, and Mobile Radio Comm.,* pp. 1073-1077, Sept. 2002.
[5] M. Nakamura, Y. Awad, and S. Vadgama, "Adaptive Control of Link Adaptation for High Speed Downlink Packet Access (HSDPA) in W-CDMA," *Proc. Int'l Symp. Wireless Personal Multimedia Comm.,* pp. 382-386, Oct. 2002.
[6] R. Kwan, P. Chong, and M. Rinne, "Analysis of the Adaptive Modulation and Coding Algorithm with Multicode Transmission," *Proc. IEEE Vehicular Technology Conf.,* pp. 2007-2011, Sept. 2002.
[7] T.E. Kolding, F. Frederiksen, and P.E. Mogensen, "Performance Aspects of WCDMA Systems with High Speed Downlink Packet Access (HSDPA)," *Proc. IEEE Vehicular Technology Conf.,* pp. 477-481, Sept. 2002.
[8] S. Abedi and S. Vadgama, "Hybrid Genetic Packet Scheduling and Radio Resource Management for High Speed Downlink Packet Access," *Proc. Int'l Symp. Wireless Personal Multimedia Comm.,* pp. 1192-1196, Oct. 2002.
[9] W.S. Jeon, D.G. Jeong, and B. Kim, "Design of Packet Transmission Scheduler for High Speed Downlink Packet Access Systems," *Proc. IEEE Vehicular Technology Conf.,* pp. 1125-1129, May 2002.
[10] Y. Ofuji, A. Morimoto, S. Abeta, and M. Sawahashi, "Comparison of Packet Scheduling Algorithms Focusing on User Throughput in High Speed Downlink Packet Access," *Proc. IEEE Int'l Symp. Personal, Indoor, and Mobile Radio Comm.,* pp. 1462-1466, Sept. 2002.
[11] Q. Zhang and H.-J. Su, "Methods for Preventing Protocol Stalling in UMTS Radio Link Control," *Proc. IEEE Int'l Conf. Comm.,* pp. 2246-2250, May 2003.
[12] A. Morimoto, S. Abeta, and M. Sawahashi, "Performance of Fast Cell Selection Coupled with Fast Packet Schedulling in High-Speed Downlink Packet Access," *Proc. IEICE Trans. Comm.,* pp. 2021-2031, 2002.
[13] L. Davis, D. Garrett, G. Woodward, M. Bickerstaff, and F. Mullany, "System Architecture and ASICs for a MIMO 3GPP-HSDPA Receiver," *Proc. IEEE Vehicular Technology Conf.,* pp. 818-822, Apr. 2003.
[14] A. Hottinen, J. Vesma, O. Tirkkonen, and N. Nefedov, "High Bit Rates for 3G and Beyond Using MIMO Channels," *Proc. IEEE Int'l Symp. Personal, Indoor, and Mobile Radio Comm.,* pp. 854-858, Sept. 2002.
[15] P. Lin, B. Lin, and I. Chlamtac, "Overflow Control for UMTS High-Speed Downlink Packet Access," *Proc. IEEE Trans. Wireless Comm.,* vol. 3, no. 2, pp. 524-532, Mar. 2004.
[16] M. Chatterjee, G.D. Mandyam, and S.K. Das, "Fast ARQ in High Speed Downlink Packet Access for WCDMA Systems," *Proc. European Wireless Conf.,* pp. 451-457, Feb. 2002.
[17] J. Zhang, W. Cao, M. Peng, and W. Wang, "Investigation of Hybrid ARQ Performance for TDD CDMA HSDPA," *Proc. IEEE Vehicular Technology Conf.,* pp. 2721-2724, Apr. 2003.
[18] A. Das, F. Khan, A. Sampath, and H.-J. Su, "Adaptive, Asynchrounous Incremental Redundancy (A^2IR) Fixed Transmission Time Intervals (TTI) for HSDPA," *Proc. IEEE Int'l Symp. Personal, Indoor, and Mobile Radio Comm.,* pp. 1083-1087, Sept. 2002.
[19] R. Love, B. Classon, A. Ghosh, and M. Cudak, "Incremental Redundancy for Evolutions of 3G CDMA Systems," *Proc. IEEE Vehicular Technology Conf.,* pp. 454-458, May 2002.
[20] A. Das, F. Khan, A. Sampath, and H.-J. Su, "Performance of Hybrid ARQ for High Speed Downlink Packet Access in UMTS," *Proc. IEEE Vehicular Technology Conf.,* pp. 2133-2137, Oct. 2001.
[21] P. Frenger, S. Parkvall, and E. Dahlman, "Performance Comparison of HARQ with Chase Combining and Incremental Redundancy for HSDPA," *Proc. IEEE Vehicular Technology Conf.,* pp. 1829-1833, Oct. 2001.
[22] G. Manuel and M. Rinne, "Performance of the Medium Access Control Protocol for the High Speed Downlink Packet Access," *Proc. IASTED Int'l Conf. Comm. Systems and Networks,* pp. 42-47, Sept. 2003.
[23] G. Manuel and M. Rinne, "Analysis of the Transmission Window for the Delay Performance of the High Speed Downlink Packet Access Protocol," *Proc. Int'l Conf. Software, Telecomm., and Computer Networks,* pp. 566-571, Oct. 2003.
[24] 3GPP TSG-RAN WG2 R2-020945, "ACK/NACK Power Offsets in Case of Realistic Channel Estimation," May 2002.
[25] TSG RAN R2-021343, "LS on HARQ ACK/NACK Error Eequirements for HSDPA," May 2002.
[26] 3GPP TSG-RAN WG2 R2-021590, "Enhancements to Stall Avoidance Mechanism," June 2002.
[27] 3GPP TS 25.308 V5.2.0, "High Speed Downlink Packet Access (HSDPA) Overall Description," Mar. 2002.
[28] 3GPP TS 25.321 V5.0.0, "Medium Access Control (MAC) Protocol Specification," Mar. 2002.
[29] 3GPP WG2-30 R2-021725, "Stall Avoidance Schemes in HARQ Entity," June 2002.
[30] 3GPP WG2-31 R2-021974, "Stall Avoidance Schemes in HARQ Entity," Aug. 2002.
[31] 3GPP TSG-RAN WG2 R2-A010016, "Dual-Channel Stop-and-Wait HARQ," Jan. 2001.
[32] 3GPP TR 25.855 V5.0.0, "High Speed Downlink Packet Access Overall UTRAN Description," Sept. 2001.
[33] H. Lin and S.K. Das, "Performance Study of Link Layer and MAC Layer Protocols to Support TCP in 3G CDMA Systems," *IEEE Trans. Mobile Computing,* vol. 4, no. 5, pp. 489-502, Sept./Oct. 2005.

Li-Chun Wang received the BS degree from National Chiao Tung University, Taiwan, ROC, in 1986, the MS degree from National Taiwan University in 1988, and the MsSci and PhD degrees from the Georgia Institute of Technology, Atlanta, in 1995, and 1996, respectively, all in electrical engineering. From 1990 to 1992, he was with the Telecommunications Laboratories of the Ministry of Transportations and Communications in Taiwan (currently the Telecom Labs of Chunghwa Telecom Co.). In 1995, he was affiliated with Bell Northern Research of Northern Telecom, Inc., Richardson, Texas. From 1996 to 2000, he was with AT&T Laboratories, where he was a senior technical staff member in the Wireless Communications Research Department. Since August 2000, he has been an associate professor in the Department of Communication Engineering of National Chiao Tung University in Taiwan. His current research interests are in the areas of cellular architectures, radio network resource management, cross-layer optimization, and cooperation wireless communications networks. Dr. Wang was a corecipient (with Gordon L. Stuer and Chin-Tau Lea) of the 1997 IEEE Jack Neubauer Best Paper Award for his paper "Architecture Design, Frequency Planning, and Performance Analysis for a Microcell/Macrocell Overlaying System," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 4, pp. 836-848, 1997 (best systems paper published in 1997 by the IEEE Vehicular Technology Society). He holds one US patent and has three more pending. Currently, he is serving as an associate editor for the *IEEE Journal of Selected Areas on Communications* wireless communications series.

Chih-Wen Chang received the BS and MS degrees in electrical engineering from National Sun Yat-Sen University, Kaohsiung, Taiwan, in 1998 and 2000, respectively. He obtained the Minor MS degree in applied mathematics and the PhD degree in communication engineering from National Chiao-Tung University HsinChu, Taiwan, in 2005 and 2006, respectively. He was awarded the IEEE student travel grant for ICC '06 and the membership of the Phi Tau Phi scholastic honor society in 2006. Since August 2006, he has been an assistant professor in the Institute of Computer and Communication Engineering at National Cheng Kung University in Taiwan. His current research interests include wireless communications, wireless networks, and cross-layer design.