行政院國家科學委員會專題研究計畫　期中進度報告

子計畫二:無線封包網路之資源管理技術(1/2)

計畫類別： 整合型計畫
計畫編號： NSC92-2219-E-009-026-
執行期間： 92 年 08 月 01 日至 93 年 07 月 31 日
執行單位： 國立交通大學電信工程學系

計畫主持人： 王蒞君

報告類型： 完整報告
報告附件： 出席國際會議研究心得報告及發表論文
處理方式： 本計畫可公開查詢

中　華　民　國 93 年 5 月 25 日

行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※
※ ※
※ B3G 無線接取網路之無線資源管理技術---子計劃二 ※
※ 無 線 封 包 網 路 之 資 源 管 理 技 術 (2 / 2) ※
※ ※
※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：□個別型計畫　　▒整合型計畫
計畫編號：NSC92-2219-E009-026
執行期間：92 年 8 月 1 日至 93 年 7 月 31 日

計畫主持人：王蒞君 副教授

▒期中報告　　□期末(總)報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

執行單位：國立交通大學電信系

中 華 民 國 93 年 5 月 31 日

摘要

停滯防止機制(stall avoidance mechanism)的目的在於減少傳輸延遲並保持媒體存取控制(MAC: medium access control)層的序列資料順暢的傳送的上層。這份報告內容包含了三個主要貢獻。第一:我們推導出寬頻分碼多重存取(WCDMA: wide band code division multiple access)系統中高速向下連聯結封包存取(HSDPA: high speed downlink packet access)現有的三種停滯防止機制之平均間空處理時間公式。三種機制分別為以時間基準停滯防止機制(timer-based stall avoidance mechanism)，視窗基準停滯防止機制(window-based stall avoidance mechanism)以及指示器基準停滯防止機制(indicator-based stall avoidance mechanism)。第二：我們建意了一個簡易的方式以加強指示器基準停滯防止機制的性能。透過分析與模擬，指示器基準停滯防止機制在平均間空處理時間上的性能表現優於其它二者停滯防止機制。再者，我們發現指示器基準停滯防止機制的加強版能有效的提高短期流通量(short-term throughput)。最後透過跨層級的模擬(cross-layer simulation)，我們發現了多通道暫時並等待(multi-channel SAW: multi-channel stop-and-wait)混合自動重傳要求(HARQ: hybrid auto-retransmission request)的通道數目設計準則。我們建議的準則是：多通道暫時並等待混合自動重傳要求的週期必須比瑞里衰減通道(Rayleigh fading channel)的同調時間(coherent time)長。

Contents

List of Figures

List of Tables

Summary

Stall avoidance mechanisms aim to reduce transmission delays and keep in-sequence delivery of medium access control (MAC) layer data to the upper layer. This report includes three major contributions. First, we derive the closed-form expressions for the average gap-processing time of the three current stall avoidance mechanisms, including the timer-based, the window-based, and indicator-based schemes, for the wideband code division multiple access (WCDMA) system with high speed downlink packet access (HSDPA). Second, we suggest a simple enhancement technique with respect to the indicator-based method. Through analysis or simulation, our results show that the indicator-based stall avoidance mechanism outperforms the timer-based and the window-based stall avoidance schemes in terms of gap-processing time. Furthermore, we find that the enhanced indicator-based stall avoidance mechanism can further improve the short-term goodput performance over the traditional indicator-based method. Last, but not the least, through a physical/MAC cross-layer investigation, we obtain some important insights when designing the multi-channel stop-and-wait (SAW) hybrid automatic repeat request (HARQ) MAC protocol. Specifically, we suggest that the number of parallel processes in the multi-channel SAW HARQ for the indicator-based method can be determined by the following rule: *(The number of parallel processes in the multi-channel SAW HARQ)*$\approx$*(The coherence time of a Rayleigh fading channel)/(Transmission time interval per process).*

Chapter 1

Introduction

High speed downlink packet access (HSDPA) is becoming an important feature for the next generation wireless data networks. The third generation partnership project (3GPP) is also specifying the HSDPA standard for the wideband code division multiple access (WCDMA) system [1]. The goal of HSDPA in the WCDMA system is to deliver mobile data services at data rates up to 10 Mbits/sec [2,3]. The key enabling technologies for HSDPA in the WCDMA system consist of physical layer fast link adaptation (e.g. adaptive modulation and coding) [4–6], fast packet scheduling [7–11], fast cell selection [12], multiple input multiple output (MIMO) antenna processing [13,14], and buffer overflow control [15]. In this report, we investigate the stall avoidance techniques to enhance the performance of fast hybrid automatic repeat request (HARQ) mechanisms in wireless channels.

The stall issue is defined as the situation when the receiver is waiting for a retransmitted packet, whereas the transmitter believes that the previous packet has already successfully reached the receiver and will never transmit that packet again. When the receiver detects a corrupted packet, it will send the negative acknowledgement (NACK) control packet to the transmitter. However, if NACK is changed to an acknowledgement (ACK) control packet due to transmission errors, the transmitter may mistakenly believe that the data packet has been successfully received, and removes the copy of this packet from its transmission buffer. Consequently, the transmitter will never send

this packet again in the future, thereby keeping the receiver to wait that lost packet hopelessly. It has been reported that the probability of the NACK signal becoming the ACK signal may be as high as 10^{-2} in a wireless channel [16]. Thus, it is crucial to resolve the stall issue in order to keep in-sequence delivery and reduce the downlink transmission delay for wireless data networks.

The multi-channel stop-and-wait (SAW) HARQ has been used for HSDPA in the WCDMA system to enhance channel utilization [17–21]. However, the stall issue may cause a long delay in the HARQ retransmission mechanism. To resolve the stall issue in the multi-channel 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 [16]. The second direction is to design stall avoidance mechanisms to inform the receiver to stop waiting for the lost MAC layer packets and forwarding all the received in-sequence packets to the upper layers [22–25]. In this case, the lost MAC layer packets can be retransmitted by the upper layer protocols. In [22], a timer-based stall avoidance mechanism was introduced, which triggers the process of forwarding received packets to the upper layer as long as a gap or a hole of the received packets' sequences in the HARQ reordering buffer lasts over a predetermined expiration period. In [23], a window-based stall avoidance mechanism was proposed to detect the stall situation in the reordering buffer through a sliding window before the timer expires. In [24, 25], the indicator-based stall avoidance mechanism introduced the concept of new data indicator (NDI) to monitor the activity of each HARQ process. When the NDI information indicates that all the HARQ processes transmit new packets, the system activates the process of forwarding received packets to the upper layer even with gaps in the reordering buffer. To our knowledge, the performances of the above stall avoidance mechanisms were evaluated only by simulation in the literature. Thus, we are motivated to evaluate the performance of these stall avoidance mechanisms through an analytical approach.

This report includes three major contributions. The first contribution is to develop an analytical framework and suggest appropriate performance metrics to evaluate the

stall avoidance mechanisms, including the timer-based, the window-based, and the indicator-based methods. We derive the closed-form expressions for the gap-processing time (defined in Chapter III) of the three considered stall avoidance mechanisms in the additive white Gaussian noise (AWGN) channel. The second contribution of this report is to propose a simple enhancement to the indicator-based stall avoidance mechanism. The objective of the proposed enhanced indicator-based stall avoidance mechanism is to ensure the timely delivery of data burst even though a NACK-to-ACK error occurs in the later portion of a data burst. Third, through a physical/MAC cross-layer investigation, we find that for the indicator-based mechanism, the cycle duration in the multi-channel SAW HARQ had better be equal to than the coherence time of the Rayleigh fading channel. Thus, with respect to the indicator-based stall avoidance mechanism, we suggest a design rule to determine that the number of parallel processes in the multi-channel SAW HARQ: *(The number of parallel processes in the multi-channel SAW HARQ)*$\approx$*(The coherence time of a Rayleigh fading channel)/(Transmission time interval per process)*.

The rest of this report is organized as follows. In Chapter II, we describe the stall issue in the multi-channel SAW HARQ mechanism. Chapter III defines two new performance metrics — gap processing time and gap-oriented throughput — to characterize the performance of stall avoidance mechanisms. Chapter IV introduces three kinds of stall avoidance mechanisms (the timer-based, the window-based, and the indicator-based schemes) and proposes an enhanced indicator-based stall avoidance mechanism. In Chapter V, we derive the closed-form expression for the average gap processing time of the three considered stall avoidance mechanisms in the AWGN channel. Chapter VI shows the performance of the stall resolution mechanisms in both the AWGN and the Rayleigh fading channels. Chapter VII gives our concluding remarks. Chapter VIII proposes some future work.

Chapter 2

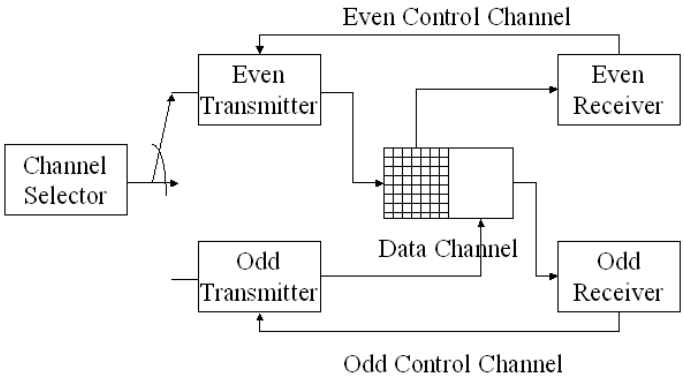
The Stall Issue in the Multi-Channel SAW HARQ

2.1 Multi-Channel SAW HARQ

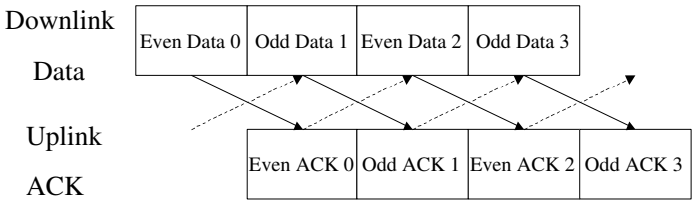
The multi-channel SAW HARQ is adopted to support the HSDPA service for the WCDMA system [1]. The basic idea of the multi-channel SAW HARQ is to implement the concept of *"keeping the pipe full"*. Figure 2.1(a) illustrates an example of a dual-channel SAW HARQ consisting of an even transmitter and an odd transmitter [26]. As shown in Fig. 2.1(b), the even transmitter and the odd transmitter send data alternatively. After sending packet 0, the even transmitter will wait 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 fully utilizes the channel capacity, thereby achieving high throughput.

2.2 An Example of the Stall Issue

Although the multi-channel SAW HARQ mechanism can efficiently utilize the channel capacity, one important concern to apply this technique in the wireless channel is the stall issue. As described previously, the stall issue is defined as the dilemma when



(a) Structure



(b) Timeline

Figure 2.1: The structure and timeline for the dual-channel SAW H-ARQ mechanism.

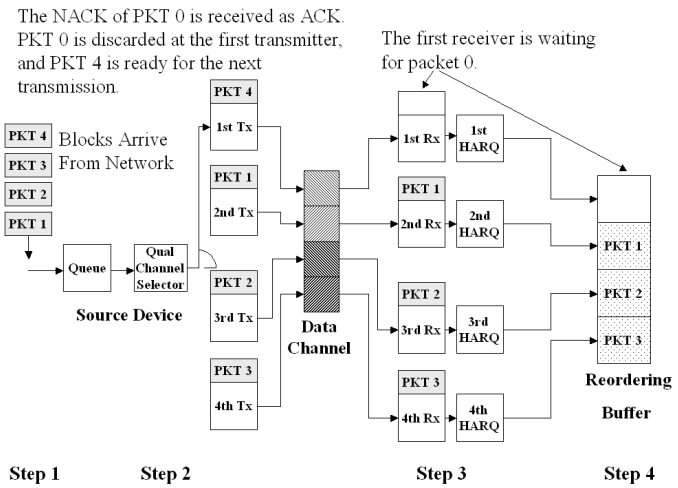


Figure 2.2: A example of the stall issue in a qual-channel SAW H-ARQ, where packet 0 is lost and packets 1, 2, and 3 are successfully received in the reordering buffer.

the receiver waits for a missing packet that will be no longer retransmitted by the transmitter. Obviously, the stall of delivering the MAC layer data to the upper layer may cause serious problems especially when sending delay-sensitive traffic. Figure 2.2 shows such an example in a qual-channel SAW HARQ. As shown in the figure, because packet 0 is lost, the first receiver sends a NACK to the first transmitter through a feedback channel. Assume a NACK-to-ACK error occurs. Then the first transmitter start sending packet 4 because it believes that packet 0 has already successfully reached the destination and thus removes the copy of packet 0 from its transmission buffer. In this situation, the first receiver will wait for packet 0 forever since it will never be sent by the first transmitter.

Chapter 3

Performance Metric

In this chapter, we define two new performance metrics to quantify the impact of the stall issue on the multi-channel SAW HARQ for the HSDPA in the WCDMA system.

3.1 Average Gap Processing Time

In this report, we call a *gap* as an idle space reserved for a lost packet in the reordering buffer at the receiver. We define two types of gap. *Type-I gap* is defined as the lost packet that can still be possibly recovered in future retransmissions, while *Type-II gap* is the one that will never be sent again by the transmitter 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.

We define *gap-processing time* as the life time of a Type-II gap, i.e. the duration when a Type-II gap appears from the reordering buffer until the receiver confirms the gap belongs to a Type-II gap. When a Type-II gap is detected, the available packets in the reordering entity must be flushed out to the upper layer even with a gap. 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 mechanism to recognize the occurrence of a Type-II gap to expedite data transmissions.

3.2 Gap-oriented goodput

We propose another new performance metric, *gap-oriented goodput*, to characterize the short-term goodput performance under the influence of Type-II gaps. Consider a data burst with at least one Type-II gap in the reordering buffer of a receiver. With respective to this data burst, the gap-oriented goodput (denoted as Γ_{gap}) is defined as

$$\Gamma_{gap} = \frac{T_p N_s}{T_{hold}} \ , \tag{3.1}$$

where T_p is the packet duration, N_s is the number of successfully received packets within a data burst, and T_{hold} is the holding period starting from the first successfully received packet until the entire data burst are forwarded to the upper protocol layer. The gap-oriented goodput is especially useful when a Type-II gap occurs in the later portion of a data burst. For example, assume that for a NACK-to-ACK error occurs in the last packet of a burst of 300 packets. Let the gap-processing time for this Type-II gap be 100 T_p and other 299 packets be successfully received (i.e. $N_s = 299$). Then from (3.1), the gap-oriented goodput is $\Gamma_{gap} = \frac{299 \times T_p}{400 T_p} = 74.75\%$. For comparison, based on a traditional definition, the goodput for this particular data burst is $\frac{299}{300} = 99.67\%$. From this example, one can see that the occurrence of a Type-II gaps can significantly affect the short-term goodput, which may not be easily captured by the traditional goodput definition.

Chapter 4

The Stall Avoidance Mechanisms

Because a NACK control signal may be corrupted during transmissions in a wireless channel, many stall avoidance mechanisms are proposed to prevent the receiver from being idle infinitely due to waiting for a packet that will never be retransmitted. In this chapter, we discuss three current stall avoidance mechanisms, the timer-based [22], the window-based [23], and the indicator-based mechanisms [24, 25]. Furthermore, we propose an enhanced indicator-based stall avoidance mechanism.

4.1 Timer-Based Mechanism

A timer-based stall avoidance approach was introduced in [22]. The basic principles of the timer-based method of [22] are described as follows:

1. Once a gap (either Type-I or Type-II) appears in the reordering buffer of the receiver, a timer is triggered.

2. When the timer expires, the reordering buffer stops waiting for the lost packet and recognize the lost packet as a Type-II gap. Hence, the receiver starts forwarding all the in-sequence available data to the upper layer.

3. If the missing packet is successfully retransmitted before the timer expires, the timer is reset.

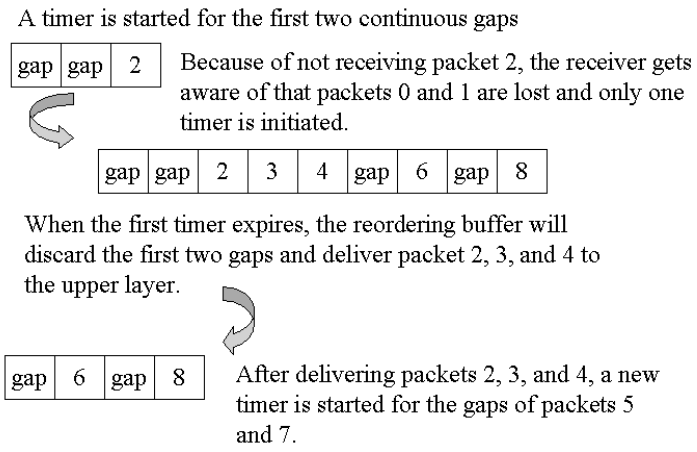


Figure 4.1: An example of the timer-based stall avoidance mechanism.

4. If a series of gaps occurs consecutively, only one timer is triggered for the first gap.

5. If additional gaps occur after the timer is triggered, the new timer will not be initiated until the previous timer expires.

Figure 4.1 illustrates the operation principles of the timer-based stall avoidance mechanism. As shown in the figure, the receiver is aware that packets 0 and 1 are missing when packet 2 is successfully received. As a result, a timer is triggered for the missing packets 0 and 1. Assume that packets 3, 4, 6 and 8 successfully reach the receiver, but packets 5 and 7 are damaged. The timer for the missing packet 5 will not be initiated until the timer for the missing packets 0 and 1 expires. As soon as the timer for packets 0 and 1 expires, packets 2, 3, and 4 will be sent to the upper radio link control (RLC) layer, which can start the RLC retransmission process for packets 0 and 1.

4.2 Window-Based Mechanism

A window-based stall avoidance mechanism [23] was proposed to enhance the performance of the timer-based stall avoidance mechanism. The window-based stall avoidance mechanism can remove Type-II gaps in the reordering queue before the timer expires. To begin with, we define the *detection window* as a set of packets for which a receiver is willing to *wait* or has already received. Differently, the traditional receiving window corresponds to a set of packets that are expected to arrive at the receiver. The detection window size can be smaller than the receiving window size. Likewise, the detection window operates as a sliding window protocol, but the detection window is activated only when a gap occurs in the reordering buffer of a receiver. If a packet is missing, the detection window size is initiated to be one. Next, there are two possible scenarios. On the one hand, if the missing packet belongs to a Type-I gap, the detection window can shrink from trailing edge when the missing packet is successfully received. On the other hand, if the missing packet belongs to a Type-II gap, the trailing edge of the detection window will be halted. The detection window will expand from leading edge as the subsequent packets arrive at the receiver. If a Type-II gap appears from the reordering buffer, the detection window size will eventually reach a pre-determined maximum threshold. We call the detection window in a *fully-booked* status when a new arriving packet finds that all available slots/seats of the detection window have been occupied by either previously successfully received packets or some missing packets. Note that the detection window size is usually designed to be large enough to guarantee a successful retransmission for a missing packet. Hence, when the detection window is fully-booked, the gap usually belongs to a Type-II gap. Consequently, as long as the detection window is fully booked, the receiver can start removing the Type-II gap by forwarding the received in-sequence packets to the upper layer before the timer expires.

Figure 4.2 illustrates the operation principles of the window-based stall avoidance mechanism. In this example, the detection window size is equal to seven and the reordering buffer contains packets 2, 3, 4, and 6. In this case, packets 0, 1, and 5 are

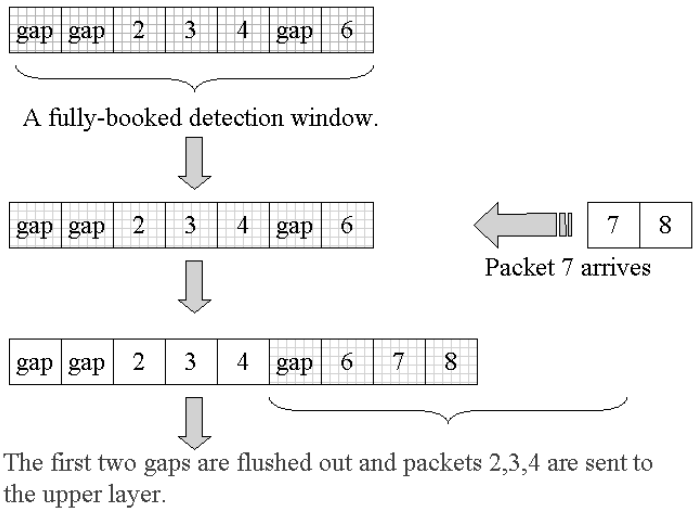


Figure 4.2: An example of the window-based stall avoidance mechanism with the detection window size equal to seven.

missing. Since all the seats of the detection window have been assigned to packets 0 to 6, the detection window is in a *fully-booked* status in the case. If no other new packet arrives, the receiver can still have a luxury to wait for the missing packets 0, 1, and 5. However, if packets 7 and 8 arrive at this moment, the receiver has an obligation to accept more packets and has to sentence the missing packets 0 and 1 to be Type-II gaps. Hence, the receiver ignores the missing packets 0 and 1 and starts forwarding the received packets 2, 3, 4 to the upper layer. After that, the trailing edge of the detection window is slid until the gap for packet 5. With packets 7 and 8 being accommodated in the reordering buffer, the detection window is now reserved by the missing packet 5 and the successfully received packets 6, 7, and 8.

4.3 Indicator-Based Mechanism

An indicator-based stall avoidance scheme was introduced in [24,25], which utilizes the new data indicator (NDI) to monitor the activity of each process in the multi-channel SAW HARQ mechanism. The NDI is a one-bit tag to indicate whether the packet is

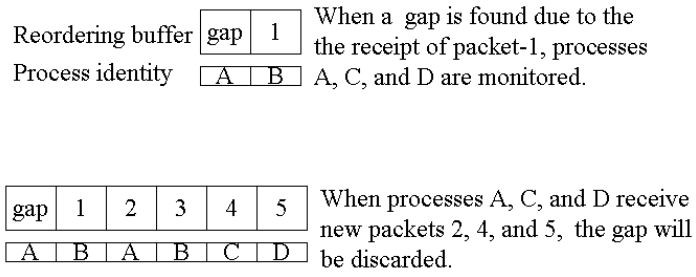


Figure 4.3: An example of the indicator-based stall avoidance mechanism for a qual-channel SAW HARQ.

a new packet or a retransmitted one. When an HARQ process sends a new packet, the NDI for that process is changed from the previous state. For the retransmitted packets, the NDIs are kept the same as the previous state. The NDI information can be transmitted in the high speed downlink control channel (HS-DCCH). Once a gap appears in the reordering buffer, the indicator-based stall avoidance mechanism starts checking each HARQ process. For a process that has changed NDI or receives a packet successfully, this implies that this process will never transmit the missing packet again. When the receiver believes that all processes will not transmit the missing packet, it can judge the missing packet belongs to a Type-II gap, and start forwarding the received in-sequence packets to the upper layer.

Figure 4.3 shows an example of the indicator-based stall avoidance mechanism for a qual-channel SAW HARQ. In this example, process A has a Type-II gap and process B is receiving packet 1. At this moment, the indicator-based mechanism starts monitoring the activity of processes A, C, and D. If all the processes A, C, and D have changed NDI or receive packets successfully, the indicator-based stall mechanism judge the missing packet in process A is a Type-II gap and start forwarding the data in the reordering queue to the upper layer when neither the timer expires nor the detection window is fully-booked.

4.4 Enhanced Indicator-Based Mechanism

We propose a simple enhancement for the indicator-based stall avoidance mechanism. This enhancement technique is especially useful when a Type-II gap occurs in the later portion of a data burst. For example, assume that the last packet in a burst of data is lost. In this situation, there will be no subsequent packets. According to the indicator-based or the window-based stall avoidance mechanism, if no subsequent packets arriving, the receiver may have a difficulty determining whether the missing packet belongs to a Type-II gap. Hence, the receiver may keep waiting for this packet until the timer expires.

To cope with this issue, we suggest the following strategy:

1. We first assume that the receiver can know the end of a burst of packets through a control channel information.

2. If a gap occurs in a later portion of a data burst, the waiting time for these missing packets is set to a relatively short period of time, i.e. 2 TTIs in our simulation.

3. If this short timer expires and the gap in the later portion of a data burst has not yet be removed, all the received packets in the reordering buffer will be forced to send to the upper layer.

There are some other enhancement techniques proposed for the indicator-based stall avoidance mechanism, such as delay-timer, a priority queue ID, and flushing indication [27]. Our proposed enhancement technique is easy to implement, which only requires the information of length of a data burst.

Chapter 5

Analysis

In this chapter, we analyze the average gap processing time performance of the timer-based, the window-based, and the indicator-based stall avoidance mechanisms.

5.1 Timer-based Stall Avoidance Mechanism

Proposition 1: *Assume that a Type-II gap occurs in a single user case. Denote P_s as the probability of a packet being successfully received and P_{N2A} as the probability of a NACK-to-ACK event. Then the average gap processing time for the timer-based stall avoidance mechanism (denoted as GPT_{timer}) can be expressed as*

$$\overline{GPT}_{timer} = \sum_{\ell=0}^{Z} \overline{GPT}(\ell) \ , \tag{5.1}$$

where

$$\overline{GPT}(\ell) = \begin{cases} (1-P_G)^Z [Z + \sum_{i=1}^{N} i(1-P_s)^{i-1} P_s] & , \quad \ell = 0 \\ P_G^{\ell} (1-P_G)^{Z-\ell} \sum_{t_1=1}^{Z-\ell+1} \sum_{t_2=t_1+1}^{Z-\ell+2} \cdots \sum_{t_\ell=t_{\ell-1}+1}^{Z} \sum_{j=1}^{\ell} \frac{2Z - t_j + \epsilon_2}{\ell} & , \quad \ell = 1, \cdots, Z \end{cases} \tag{5.2}$$

and Z is the expiry time of the timer normalized to the transmission time interval (TTI) of a packet.

Note that in (5.2) P_G is the probability of having a Type-II gap, equal to

$$P_G = (1 - P_s)P_{N2A} \ . \tag{5.3}$$

Furthermore, t_j is the duration from the beginning of the previous timer until the end of Gap_j $(j = 1, \cdots, \ell)$, and

$$\varepsilon_2 \ = \ \sum_{m=1}^{Z} \sum_{n=m}^{N} (n - m + 1)(1 - P_G)^{m-1} P_G (1 - P_S)^{(n-1)} P_S \ , \tag{5.4}$$

where N represents the duration in terms of the number of TTIs between a gap and the first successfully received packet afterwards.

Proof: To prove this propostion, we first assume that a timer is already initiated for a gap. Denote $\overline{GPT}(i)$ as the average gap-processing time with i Type-II gaps. Before the timer expires, there are the following two possible scenarios:

(I) No new Type-II gap occurs before the timer expires ($\ell = 0$):

As shown in Figure 5.1(a), Gap_0 occurs after the timer of a previous gap expires. The timer for Gap_0 will not be initiated until another packet is successfully received, say ε_1 TTIs later. In this case, the gap processing time for Gap_0 is $Z+\varepsilon_1$. From (5.3), we know that the probability of having no new Type-II gap within Z TTIs is $(1-P_G)^Z$. Now assume that the first successfully received packet after Gap_0 is N slots later. Then, the average time to initiate the timer of Gap_0 is $\sum_{i}^{N} i(1 - P_s)^{i-1} P_s$ TTIs. Thus, we have

$$\overline{GPT}(0) \ = \ (Z + \sum_{i=1}^{N} i(1 - P_S)^{(i-1)} P_S)(1 - P_G)^Z \ . \tag{5.5}$$

(II) Some new Type-II gaps occur before the timer expires ($\ell \neq 0$):

Assume that additional ℓ new Type-II gaps (denoted as $Gap_1, \cdots, Gap_\ell$) occur before the timer expires, as shown in Fig. 5.1(b). Based on the timer-based stall avoidance mechanism all these Type-II gaps can be handled only after the current timer expires. That is, Gap_j $(j = 1, \cdots, \ell)$ has to wait at least $(Z - t_j)$ before starting

its own timer. The probability of having ℓ Type-II gaps with a particular pattern can be expressed as

$$P_{Gap_\ell} = P_G^\ell (1 - P_G)^{Z-\ell} \ . \tag{5.6}$$

For simplicity, we first consider a worst case where packets between any two Type-II gaps are not received correctly. Let Gap_ℓ occur at m TTIs before the expiry time of a previous gap. Assume that the first correctly received packet after Gap_ℓ (denoted as "O" in the figure) is n TTIs later. In this case, a new timer will start at $(n-m+1)$ TTIs after the timer of the current gap expires. Note that the probability of the event that Gap_ℓ occurs at m TTIs before the timer of a previous gap expires is $(1-P_G)^{m-1}P_G$ and the probability of having $(n-1)$ consecutive missing packets after Gap_ℓ is $(1-P_s)^{n-1}P_s$. Thus, the timer for Gap_ℓ starts at

$$Z - t_\ell + \sum_{m=1}^{Z} \sum_{n=m}^{N} (n-m+1)(1-P_G)^{m-1} P_G (1-P_s)^{n-1} P_s \ . \tag{5.7}$$

Thus, the gap processing time for Gap_ℓ is equal to

$$2Z - t_\ell + \sum_{m=1}^{Z} \sum_{n=m}^{N} (n-m+1)(1-P_G)^{m-1} P_G (1-P_s)^{n-1} P_s \ . \tag{5.8}$$

Then, from (5.7) and (5.8), the average gap processing time for these ℓ Type-II gaps given $t_1, \cdots, t_\ell$ is equal to

$$\overline{Gap}(\ell | t_1, \cdots, t_\ell) = \sum_{j=1}^{\ell} \frac{2Z - t_j + \epsilon_2}{\ell} \ , \tag{5.9}$$

where

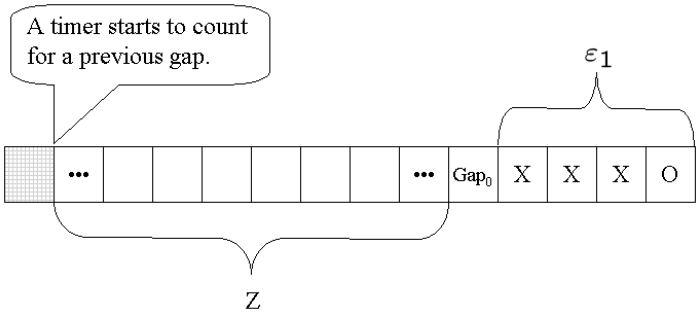
$$\epsilon_2 = \sum_{m=1}^{Z} \sum_{n=m}^{N} (n-m+1)(1-P_G)^{m-1} P_G (1-P_s)^{n-1} P_s \ . \tag{5.10}$$

Now we consider all the possible positions for ℓ Type-II gaps in Z slots. Then, we have

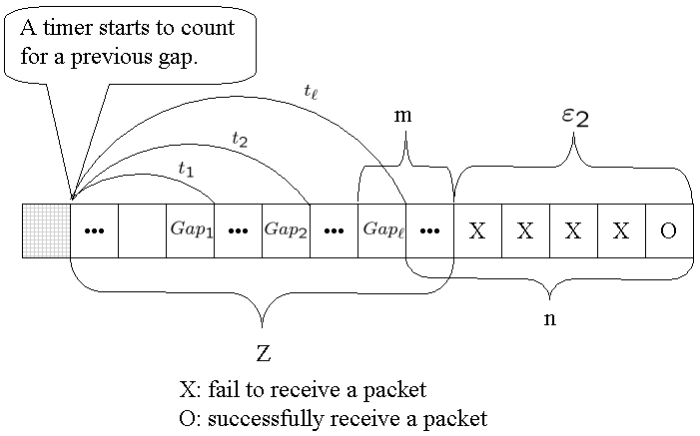
$$\overline{Gap}(\ell) = P_G^\ell (1-P_G)^{Z-\ell} \sum_{t_1=1}^{Z-\ell+1} \sum_{t_2=t_1+1}^{Z-\ell+2} \cdots \sum_{t_\ell=t_{\ell-1}+1}^{Z} \sum_{j=1}^{\ell} \frac{2Z - t_j + \epsilon_2}{\ell} \tag{5.11}$$

Combining (5.5) and (5.11), we prove Proposition 1.

■



(a) No new Type-II gap occurs before the timer expiry for a previous gap.



(b) Some new Type-II gaps occur before the timer expires for a previous gap.

Figure 5.1: Two scenarios for the timer-based stall avoidance mechanism to remove a Type-II gap.

5.2 Window-based Stall Avoidance Mechanism

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

Proposition 2: *Consider the window-based stall avoidance mechanism in an M-channel SAW HARQ having a detection window with a size of W. Assume that a Type-II gap occurs at a time slot and the maximum allowable retransmission times is L. Then, the average gap-processing time for the window-based mechanism is*

$$\overline{GPT}_{window} = M + \sum_{m=1}^{M}[(W-m)\sum_{n=1}^{L} nP_{new}P_{old}^{n-1}]P_M(m) \ , \tag{5.12}$$

where

$$P_M(m) = \binom{M-1}{m-1} P_{new}^{m-1} P_{old}^{M-m} \quad , \tag{5.13}$$

$$P_{new} = P_s + (1-P_s)P_{N2A} \ , \tag{5.14}$$

and

$$P_{old} = (1-P_s)(1-P_{N2A}) \ . \tag{5.15}$$

Proof: According to the window-based stall avoidance mechanism, a Type-II gap can be detected when the detection window is in the fully-booked status. We will prove proposition in three steps:

1. Assume that a Type-II gap occurs in a M-channel SAW HARQ. Then, the process producing this particular Type-II gap will sent a new packet (PKT^*) in the next cycle, i.e. M TTIs later. Thus, the gap-processing time of removing this Type-II gap is at least M TTIs.

2. Let PKT^* be positioned at the m-th seat of the receiving buffer, as shown in Fig. 5.2. This implied that $(m-1)$ processes carry new packets and the remaining

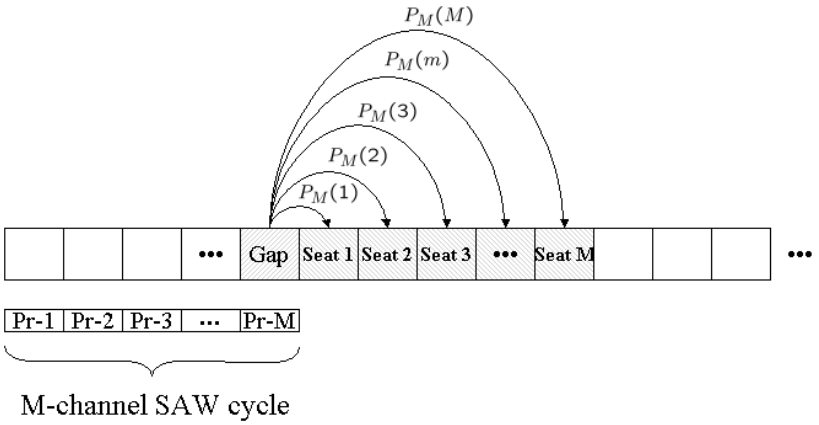


Figure 5.2: An illustration for the gap-processing time of the window-based stall avoidance mechanism.

$(M-m)$ processes retransmit old packets. Notice that the probability of PKT^* being at the m-th seat in the reordering buffer is equal to

$$P_M(m) = \binom{M-1}{m-1} P_{new}^{m-1} P_{old}^{M-m} \ , \tag{5.16}$$

where

$$\sum_{m=1}^{M} P_M(m) = 1 \ . \tag{5.17}$$

3. If PKT^* is positioned at the m-th seat, then the Type-II gap can be detected as long as the receiver recognizes the detection window of size W is in a fully-booked status. In this situation, a fully-booked status implies that the remaining $(W-m)$ seats in the receiver buffer are all occupied. Denote T_0 as the average time of the detection window extending one seat from the leading edge. Then, T_0 can be written as

$$T_o \ = \ \sum_{n=1}^{L} n P_{new} P_{old}^{n-1} \ , \tag{5.18}$$

where L is the maximum transmission times for a packet. From the above discussions, the Type-II gap will be removed after $(W-m)T_o + M$ TTIs. Note that m is a random variable with a probability mass function defined in (5.16).

From (5.16) and (5.18), the average gap-processing time for the window-based stall mechanism can be calculated as follows:

$$\overline{GPT}_{window} = M + \sum_{m=1}^{M}[(W-m)\sum_{n=1}^{L} nP_{new}P_{old}^{n-1}]P_M(m) \ . \quad (5.19)$$

■

5.3 Indicator-based Stall Avoidance Mechanism

Now we analyze the average gap-processing time of the indicator-based stall avoidance mechanism.

Proposition 3: *Consider an M-channel SAW HARQ process. The gap-processing time for the indicator-based method can be calculated as follows:*

$$\begin{aligned}\overline{GPT}_{indicator} &= M(P_{new}+P_{old}P_s)^{M-1} + \\ &\sum_{k=1}^{C}\sum_{m=2}^{M}(m-1+kM)P_{old}^{k}P_{\epsilon}(\sum_{i=0}^{k}P_{old}^{i}P_{\epsilon})^{m-2}(\sum_{j=0}^{k-1}P_{old}^{j}P_{\epsilon})^{M-m} \quad (5.20)\end{aligned}$$

where

$$P_{\epsilon} = (1-P_s)P_{N2A} + P_{old}P_s \ ; \quad (5.21)$$

C is the required cycles of involving all the processes in the M-channel SAW HARQ to remove a Type-II gap; P_{new} and P_{old} were defined in (5.14) and (5.15).

Proof: For a particular process in an M-channel SAW HARQ, we define four states depending on whether a new packet or an old packet is received successfully. Let $S_1(\beta)$ and $S_3(\beta)$ represent the probability of successfully receiving a new packet in the β-th cycle and that of receiving an old packet, respectively. Similarly, denote $S_2(\beta)$ and $S_4(\beta)$ as the failure probability of receiving a new packet in the β-th cycle and that of an old packet, respectively. Recall that if a Type-II gap appears in the reordering buffer, the indicator-based mechanism can detect this Type-II gap by checking the status of all the processes. Specifically, when all the processes are in states 1, 2, or 3,

the transmitter will no longer send the missing packet for that Type-II gap. Figure 9 shows that only in state 4, the receiver has to keep monitoring the states of that process. Clearly, we have

$$\begin{aligned} S_1(\beta) &= S_4(\beta-1)P_{N2A}Ps & (5.22)\\ S_2(\beta) &= S_4(\beta-1)P_{N2A}(1-Ps) & (5.23)\\ S_3(\beta) &= S_4(\beta-1)(1-P_{N2A})Ps & (5.24)\\ S_4(\beta) &= S_4(\beta-1)(1-P_{N2A})(1-Ps)\ , & (5.25) \end{aligned}$$

of which the initial states are

$$\begin{aligned} S_1(0) &= P_{new}P_s & (5.26)\\ S_2(0) &= P_{new}(1-P_s) & (5.27)\\ S_3(0) &= P_{old}P_s & (5.28)\\ S_4(0) &= P_{old}(1-P_s)\ . & (5.29) \end{aligned}$$

Now we consider the following two possible scenarios.

(I) The gap can be removed within one cycle:

Assume that a Type-II gap appears in the first process of cycle 0 as shown in Fig. 5.4. When all the processes except for the first process are in states 1, 2, or 3, the gap can be removed in cycle 1 of process-1. Because the gap in process-1 is assumed to be a Type-II gap, process-1 will transmit a new packet in cycle 1. When receiving this new packet from process-1, the new data indicator for process 1 switches to the "ON" state. Thus, the receiver knows that the missing packet for this Type-II gap will never be retransmitted by process 1. If all the other $(M-1)$ processes are also transmitting other packets, the receiver can recognize the missing packet is a Type-II gap. In this case, the receiver takes M TTIs to remove the Type-II gap. The probability of a process in states 1, 2 or 3 of cycle 0 is

$$\begin{aligned} S_1(0)+S_2(0)+S_3(0) &= P_{new}P_s+P_{new}(1-P_s)+P_{old}P_s\\ &= P_{new}+P_{old}P_s\ . & (5.30) \end{aligned}$$

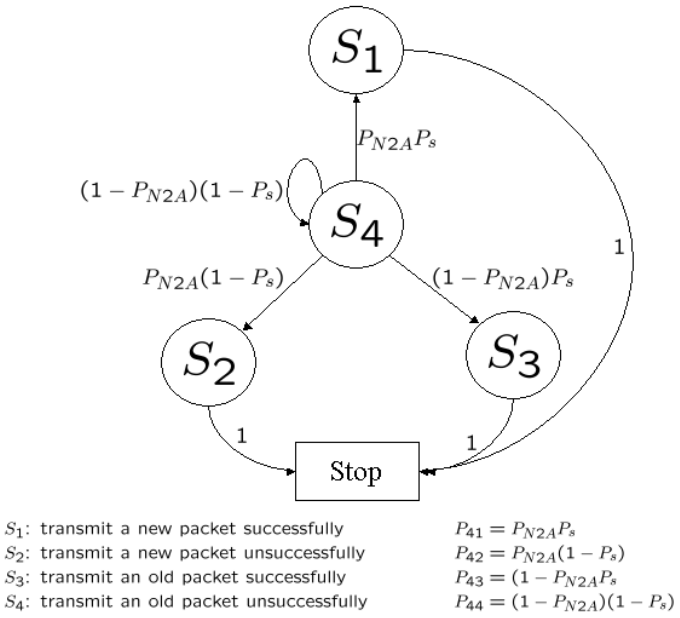


Figure 5.3: The state transition diagram for the indicator-based stall avoidance mechanism.

Thus, the gap-processing time for removing a Type-II gap within one cycle can be expressed as

$$\overline{GPT}_0 = M(P_{new} + P_{old}P_s)^{(M-1)} \ . \tag{5.31}$$

(II) The gap needs more than one cycle to be removed:

Let $P_{Ind}(\beta)$ be the probability for a process in states 1, 2, or 3 in the β-th cycle. That is $P_{Ind}(\beta) = S_1(\beta) + S_2(\beta) + S_3(\beta)$. Note that if a process is in states 1, 2, or 3, then the indicator-based stall avoidance mechanism stops monitoring the function of that process. From (5.22) to (5.24), we obtain

$$\begin{aligned} P_{Ind}(\beta) &= S_1(\beta) + S_2(\beta) + S_3(\beta) \\ &= S_4(\beta-1)P_{N2A} + S_4(\beta-1)(1-P_{N2A})Ps \ . \end{aligned} \tag{5.32}$$

By iteratively substituting (5.25) and (5.29) into (5.22)-(5.24), we can obtain

$$P_{Ind}(\beta) = P_{old}^{\beta}[(1-P_s)P_{N2A} + P_{old}P_s] \ . \tag{5.33}$$

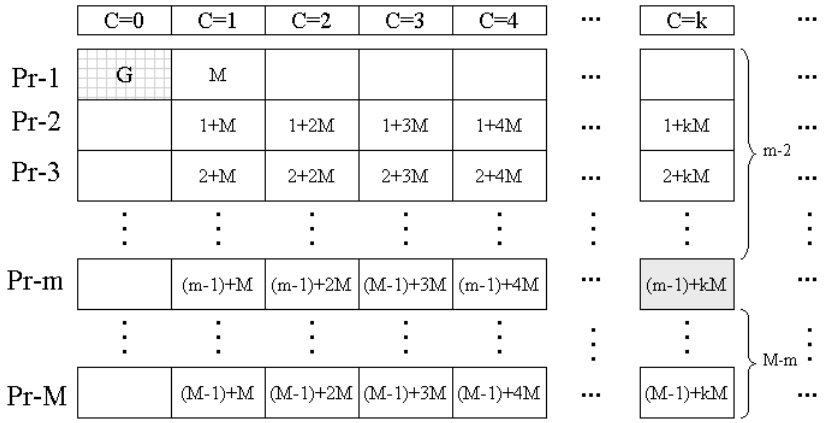


Figure 5.4: An illustration for the gap-processing time of the indicator-based stall avoidance mechanism.

Next, we calculate the probability of all the processes entering states 1, 2, or 3. Assume that the gap is removed at the k-th cycle of process m, where $m \geq 2$ and $k \geq 1$. In this case, the gap processing time is $(m-1+kM)$, as shown in Fig. 5.4. Note that different combinations of k and m are associated with different occurrence probabilities. Represent $P_{Ind}^{(\alpha)}(i_\alpha)$ as the probability of process α entering states 1, 2, or 3 in the i_α-th cycle and $P_{Ind}(m,s)$ as the probability for the Type-II gap being detected in the k-th cycle of the m-th process. Then, $P_{Ind}(m,s)$ can be expressed as

$$P_{Ind}(m,k) = \overbrace{P_{Ind}^{(2)}(i_2)P_{Ind}^{(3)}(i_3)\cdots P_{Ind}^{(m-1)}(i_{m-1})}^{(m-2)} \cdot P_{Ind}^{(m)}(k) \cdot \overbrace{P_{Ind}^{(m+1)}(j_{m+1})\cdots P_{Ind}^{(M)}(j_M)}^{(M-m)}, \tag{5.34}$$

where

$$0 \leq i_\alpha \leq k, \quad 2 \leq \alpha \leq m-1 \tag{5.35}$$

and

$$0 \leq j_\gamma \leq k-1, \quad m+1 \leq \gamma \leq M \ . \tag{5.36}$$

From (5.33), (5.35), and (5.36) we have

$$P_{Ind}^{(n)}(i_n) = \begin{cases} \sum_{i=0}^{k} (P_{old})^i P_\epsilon & , \quad 2 \le n < m \\ P_{old}^k P_\epsilon & , \quad n = m \\ \sum_{j=0}^{k-1} (P_{old})^j P_\epsilon & , \quad m+1 \le n \le M \end{cases} \tag{5.37}$$

where

$$P_\epsilon = (1 - P_s) P_{N2A} + P_{old} P_s \ . \tag{5.38}$$

From (5.34) and (5.37), the average gap-processing time for case (II) can be expressed as

$$\begin{aligned} \overline{GPT}_1 &= \sum_{k=1}^{C} \sum_{m=2}^{M} (m - 1 + kM) P_{Ind}(m, k) \\ &= \sum_{k=1}^{C} \sum_{m=2}^{M} (m - 1 + kM) P_{old}^k P_\epsilon (\sum_{i=0}^{k} P_{old}^i P_\epsilon)^{m-2} (\sum_{j=0}^{k-1} P_{old}^j P_\epsilon)^{M-m} \ , \end{aligned} \tag{5.39}$$

where C is the required cycles of involving all the processes of an M-channel SAW HARQ to remove a Type-II gap. Note that the value of C can be obtained by the following equation:

$$(P_{new} + P_{old} P_s)^{M-1} + \sum_{k=1}^{C} \sum_{m=2}^{M} P_{old}^k P_\epsilon (\sum_{i=0}^{k} P_{old}^i P_\epsilon)^{m-2} (\sum_{j=0}^{k-1} P_{old}^j P_\epsilon)^{M-m} = 1 \ ,$$

where P_{new}, P_{old}, P_s, M, and P_ϵ are all given parameters. With (5.31) and (5.39), the average gap-processing time for the indicator-based stall avoidance mechanism is equal to

$$\begin{aligned} \overline{GPT}_{indicator} &= \overline{GPT}_0 + \overline{GPT}_1 \\ &= M(P_{new} + P_{old} P_s)^{M-1} + \\ &\quad \sum_{k=1}^{C} \sum_{m=2}^{M} (m - 1 + kM) P_{old}^k P_\epsilon (\sum_{i=0}^{k} P_{old}^i P_\epsilon)^{m-2} (\sum_{j=0}^{k-1} P_{old}^j P_\epsilon)^{M-m} \end{aligned} \tag{5.40}$$

■

Chapter 6

Numerical Results

In this chapter, we investigate the performance of stall avoidance mechanisms in both the AWGN and the Rayleigh fading channels by simulations and analysis. Table 6.1 lists the simulated system parameters.

6.1 Comparison of Stall Avoidance Mechanisms in the AWGN Channel

Figure 6.1 shows the gap-processing time of the timer-based stall avoidance mechanism in the AWGN channel with different timer expiration values. First, as shown in the

Table 6.1: Simulation Parameters

N-channel SAW H-ARQ	N=4, 8, 12
Frame Size	320 bits
Error Checking	CRC
TTI	2 msec
Environment 1	AWGN
Environment 2	Rayleigh fading with f_d = 3, 10, 50, 100 Hz
Burst Data Size	150, 2000 Packets

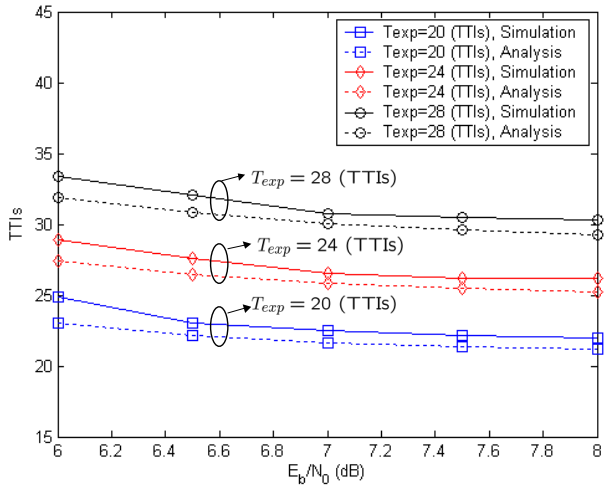


Figure 6.1: Comparison of gap-processing time of the timer-based stall avoidance mechanism with different timer expiration duration for the 4-channel SAW HARQ in the AWGN channel .

figure, the simulation results closely approach the analytic results of (5.1) in Proposition 1. For example, when $E_b/N_0 = 7$ dB and the expiry duration is $Z = 24$ TTIs, the simulation and analysis values of the average gap-processing time for the timer-based are 26.57 and 25.85 TTIs, respectively. Second, we find that the larger the expiration duration of the timer, the longer the gap-processing time. Note that $Z = 24$ is just used for validating the accuracy of the analytical results. Typically, the timer expiry can be quite long. Determining the optimal timer value is not an easy task. On the one hand, the timer needs to be long enough for a successful retransmission in the future. On the other hand, a long timer will delay the delivering of the MAC layer data to the upper layer. Hence, there exists a tradeoff between packet reliability and transmission delay. Proposition 1 can quantitatively estimate the average gap-processing time instead of conducting a time consuming simulation.

Figure 6.2 compares the gap-processing time of the window-based and the indicator-based stall avoidance mechanisms in the AWGN channel with the detection window size equal to 30 TTIs. From the figure, we find that the indicator-based mechanism outperforms the window-based mechanism. For $E_b/N_o = 7$ dB, the gap-processing

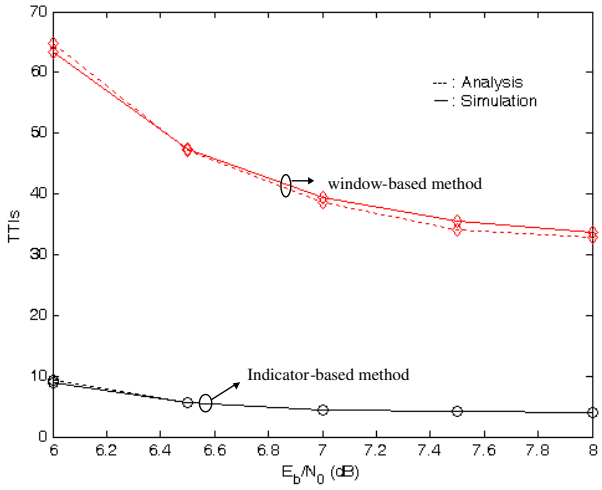


Figure 6.2: Comparison of gap-processing time of the window-based and the indicator-based stall avoidance mechanisms with a 4-channel SAW HARQ in the AWGN channel .

time for the window-based mechanism is 39.33 TTIs, while the gap-processing time for the indicator-based method is 5.87 TTIs. Note that the setting of the window size is another tradeoff design issue between the number of allowable retransmission times and the short average gap-processing time. An appropriate window size depends on the channel conditions and the size of the reordering buffer.

Figure 6.3 shows the average gap-processing time for the indicator-based stall avoidance mechanism with 4, 8, and 12-channel SAW HARQ schemes, respectively. As shown in the figure, the average gap-processing time increases as the number of processes in the SAW HARQ increases. Although the system throughput is improved by increasing the number of processes in the SAW HARQ scheme, the downside of the longer gap-processing time cannot be ignored. That is, there exists another tradeoff between system throughput and the gap-processing time for the multi-channel SAW HARQ.

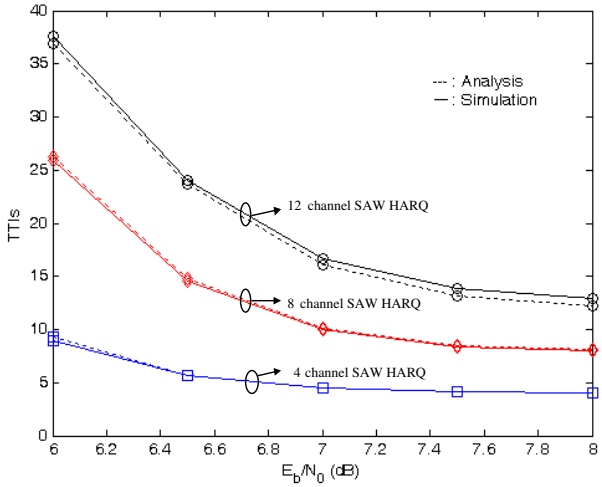


Figure 6.3: Effect of the number of processes in the multi-channel SAW HARQ on the gap-processing time for the indicator-based avoidance mechanism in the AWGN channel .

6.2 Comparison of Stall Avoidance Mechanisms in the Rayleigh Fading Channel

Figure 6.4 shows the gap-processing time of the timer-based stall avoidance mechanism in the Rayleigh fading channel with different Doppler frequencies. Let the timer expiry duration be 24 TTIs. From the figure, one can find that the average gap-processing time for the timer-based stall avoidance mechanism is longer in the Rayleigh channel with a lower Doppler frequency than that in the Rayleigh channel with a higher Doppler frequency. For example, when $\frac{E_b}{N_0} = 7$ dB, the average gap-processing time for $f_d = 10$ Hz and that for $f_d = 100$ Hz are 39 and 30.6 TTIs, respectively. This phenomenon can be explained by the fact that a channel with a lower Doppler frequency causes a lower level crossing rate, thereby yielding a higher probability of having consecutive gaps.

Figure 6.5 compares the gap-processing time for the window-based method in the Rayleigh fading channel with a window size of 30 packets and Doppler frequencies equal to 10 Hz and 100 Hz. For $\frac{E_b}{N_0} = 7$ dB, a faster varying fading channel ($f_d = 100$ Hz) increases the gap-processing time to 74 TTIs as compared to 71.45 TTIs in the case

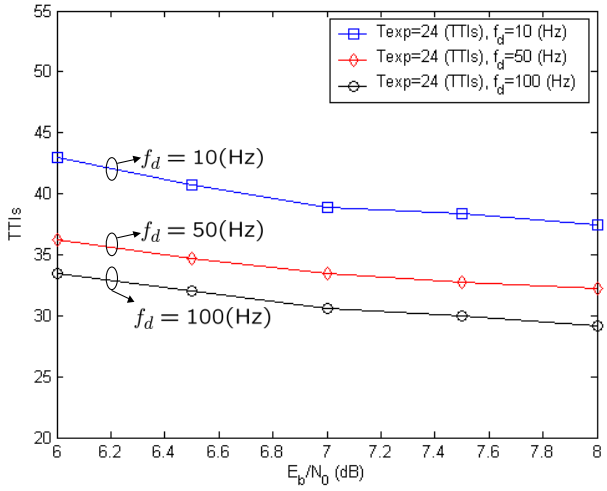


Figure 6.4: Effect of Doppler frequency on the gap-processing time for the timer-based stall avoidance mechanism with a 4-channel SAW HARQ in Rayleigh fading channels; where the Doppler frequency f_d =10, 50, and 100 Hz.

of $f_d = 10$ Hz. A faster varying fading channel usually has a higher packet error rate, thereby decreasing the probability of the detection window extending its size. That is, a faster fading channel may decrease the probability of the detection window being fully-booked. Consequently, in the Rayleigh fading channel with a higher Doppler frequency, the window-based stall avoidance mechanism takes longer time to detect a Type-II gap, which is different from the case in the timer-based stall avoidance mechanism.

In the Rayleigh fading channel with different Doppler frequencies ($f_d = 3, 10, 100$ Hz), Figs. 6.6 and 6.7 show the gap-processing time performance of the indicator-based stall avoidance mechanism for the 4-channel SAW HARQ and that for the 12-channel SAW HARQ, respectively. Comparing these two figures, we find that Doppler frequency affects the gap-processing time differently. For example, for the 4-channel SAW HARQ (Fig. 6.6), the average gap-processing time of $f_d = 100$ Hz is the shortest among the three curves, while in the 12-channel SAW HARQ case (Fig. 6.7) the gap-processing time for $f_d = 10$ Hz is the shortest. We explain this observation as follows:

1. Effect when the cycle duration is shorter than the channel coherence time:

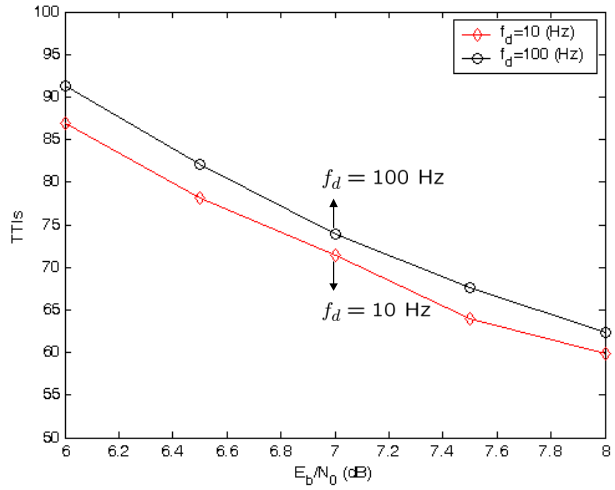


Figure 6.5: Effect of Doppler frequency on the gap-processing time for the window-based stall avoidance mechanism with a 4-channel SAW HARQ in Rayleigh fading channels.

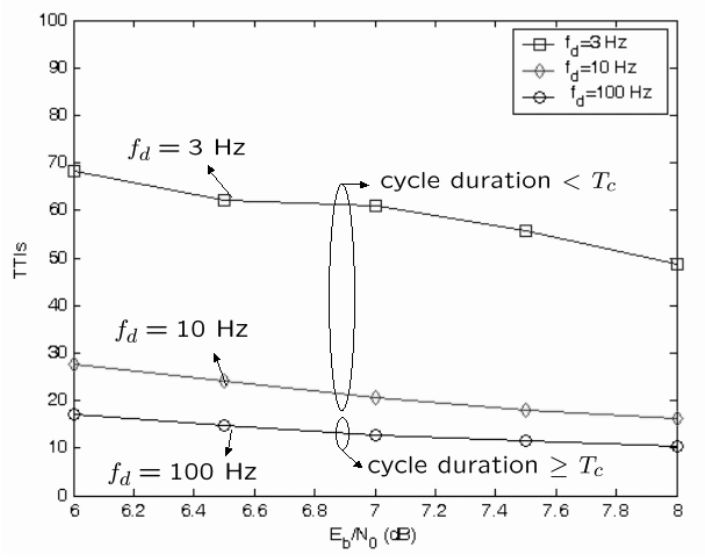


Figure 6.6: Effect of Doppler frequency and cycle duration on the the gap-processing time of the indicator-based stall avoidance mechanism with a 4-channel SAW HARQ in Rayleigh fading channels.

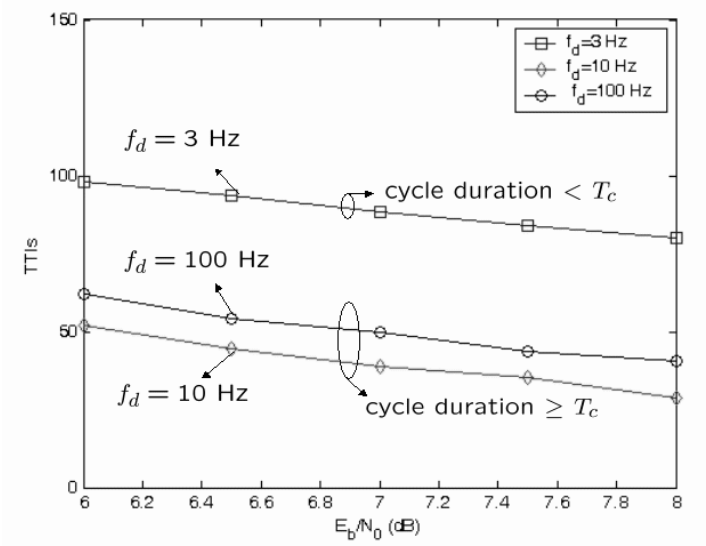


Figure 6.7: Effect of Doppler frequency and cycle duration on the the gap-processing time of the indicator-based stall avoidance mechanism with a 12-channel SAW HARQ in Rayleigh fading channels.

When the cycle duration of the multi-channel SAW HARQ is longer than the channel coherence time, all processes in the multi-channel SAW HARQ can experience different channel conditions in different cycles. Hence, a process will not stay in a bad channel condition for a long period of time. This property is useful for the receiver to improve packet error rate performance in a multi-channel SAW HARQ process. On the contrary, when the cycle duration is shorter than the channel coherence time, the multi-channel SAW HARQ processes may encounter a bad channel condition for several cycles, which will detain the receiver to receive a packet successfully. Thus, a shorter cycle duration decreases the success probability of all processes receiving packets, thereby increasing the gap-processing time. From [28], the channel coherence time can be calculated by

$$T_c = \frac{9}{16\pi f_d} \ , \tag{6.1}$$

where f_d is the Doppler frequency. The coherence time for $f_d = 3, 10, 100$ Hz are 0.0596, 0.0179, 0.00179 seconds, respectively. The cycle duration for the 4-channel and 12-channel SAW HARQ with a 2-msec TTI per process are 0.008

and 0.024 seconds, respectively. The cycle duration of the 4-channel SAW HARQ (0.008 seconds) is shorter than the coherence time (0.0596 and 0.0179 seconds) with $f_d = 3$ and 10 Hz, respectively. Hence, the gap-processing time for $f_d = 3$ and 10 Hz are worse than that of $f_d = 100$ Hz, as shown in the Fig. 6.6. In Fig. 6.7, it is also shown that the gap-processing time of $f_d = 3$ Hz is longer than that of $f_d = 10$ and 100 Hz because the cycle duration for the 12-channel (0.024 seconds) is shorter than the channel coherence time with $f_d = 3$ Hz (0.0596 secs), but longer than that with $f_d = 10$ (0.0179 secs) and that of 100 Hz (0.00179 secs).

2. Doppler effect:

   (a) Case I: *cycle duration* $<$ *channel coherence time*

   In this case, we find that the higher the Doppler frequency, the shorter the gap-processing time. In Fig. 6.6, for $E_b/N_0 = 7$ dB, the gap-processing time with f_d=10 Hz is 24.7 TTIs, and the one with $f_d = 3$ Hz is 60.8 TTIs. In this example, the changing rate of a Rayleigh fading channel with $f_d = 3$ and 10 Hz is not fast enough for the 4-channel SAW HARQ to provide different channel conditions in adjacent cycles. Compared to $f_d = 3$ Hz, the channel with $f_d = 10$ Hz has a higher probability to provide different channel conditions in different cycles. Thus, the indicator-based stall avoidance mechanism can have shorter gap-processing time in the channel with $f_d = 10$ Hz than that in the channel with $f_d = 3$ Hz.

   (b) Case II: *cycle duration* $\geq$ *channel coherence time*

   In this situation, we find that the higher the Doppler frequency, the longer will be the gap-processing time. For $E_b/N_0 = 7$ dB in Fig. 6.7, it is shown that the average gap-processing time with $f_d = 100$ Hz is 50 TTIs, and the one with $f_d = 10$ Hz is 38.7 TTIs. For the 12-channel SAW HARQ, the changing rate of a Rayleigh fading channel with $f_d = 10$ and 100 Hz is always fast enough to provide different channel conditions in different

cycles. In this situation, a higher Doppler frequency may cause a higher packet error rate because an erroneous bit may corrupt a whole packet. When the cycle duration is longer than the channel coherence time, the average gap-processing time in a channel with a higher Doppler frequency is therefore longer than that in a channel with a lower Doppler frequency.

From the above physical/MAC cross-layer investigation, we can suggest a design principle for the indicator-based stall avoidance mechanism to determine the number of processes of the multi-channel SAW HARQ as follows: (*the cycle duration of the multi-channel SAW HARQ* $\approx$ *the channel coherence time of the Rayleigh fading channel*).

Figure 6.8 compares the gap-oriented goodput performance between the enhanced indicator-based stall avoidance mechanism and the traditional indicator-based stall avoidance mechanism in a Rayleigh fading channel. We consider a data burst with 150 packets. For the case with $f_d = 100$ Hz and $E_b/N_0 = 7$ dB, we find that the gap-oriented goodput of the enhanced indicator based stall avoidance mechanism is 0.49, while the gap-oriented goodput of the traditional indicator based method is 0.37. In the considered case, the short-term goodput performance of the proposed enhanced indicator based stall avoidance mechanism is 32% better than that of the conventional indicator-based method. Note that both stall avoidance mechanisms have the similar average gap-processing time.

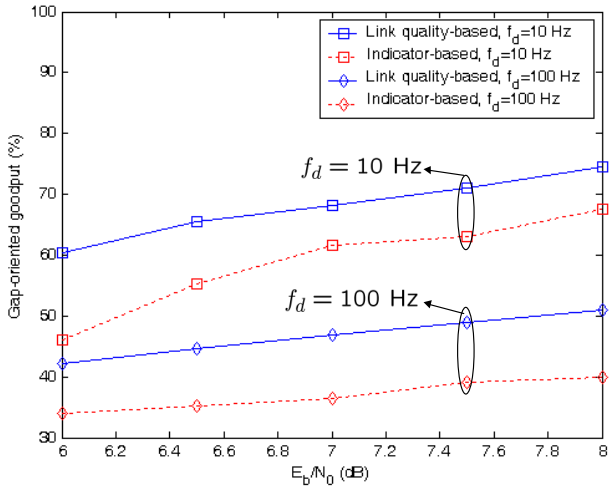


Figure 6.8: Comparison of gap-oriented goodput between the enhanced indicator-based and the traditional indicator-based stall avoidance mechanisms with a 4-channel SAW HARQ in the Rayleigh fading channel.

Chapter 7

Conclusions

In this report, we have investigated the performance of four stall avoidance mechanisms for the WCDMA system with HSDPA in both the AWGN channel and the Rayleigh fading channel. By analysis and simulation, we show that the indicator-based stall avoidance mechanism performs better than the timer-based and the window-based stall avoidance mechanism in terms of gap-processing time. Furthermore, we propose an enhanced indicator-based scheme to improve the short-term goodput performance for the current indicator-based scheme.

More importantly, we have derived analytical expressions for the gap-processing time of the considered stall avoidance mechanisms in the AWGN channel. From a physical/MAC cross-layer investigation, we observe that it is important to take the physical layer impact into account when implementing the multi-channel SAW HARQ. Specifically, we suggest that for the indicator-based stall avoidance mechanism, the number of parallel processes of the multi-channel SAW HARQ can be simply decided by the ratio of *the coherence time of a Rayleigh fading channel over the transmission time interval per process*. Interesting issues that can be extended from this work include the analysis of the multi-channel SAW HARQ in the multi-user case and in the Rayleigh fading channel.

Chapter 8

Future Work

We desire to extend the Proposition 3 of average gap-processing time to multiuser case in the Rayleigh fading channel. With the results of average gap-processing time for multiuser communications, we want to propose a new performance metric to overall take the throughput, the fairness, the queue length, and the average gap-processing time, etc., into account of scheduling policy. At last, we will propose a scheduling scheme for HSDPA.

Appendix A

Simulation Platform

A.1 The Signaling Processing Part

In order to investigate the cross-layer performance of the stall avoidance mechanisms, we develop a cross-layer simulation platform by Matlab-Simulink tool. The Simulink provides GUI interface and a flexible extensibility. This platform has integrated function to simulate closed-loop power control and stall avoidance mechanisms. The detailed will be explained in the following paragraphs. The overall picture of this platform is shown in Fig. A.1. The signaling processing part in this platform consists of several blocks which are "Binary Source", "Modulation" and "Demodulation", "Spread" and "Despread", "Scramble" and "Descramble" and "Rake".

The first block is the "Binary Source". This block is to produce binary signal for the consequent procedures. In HSDPA system, the spreading factor is fixed up to 16; therefore running a HSDPA stall avoidance mechanism we have to choose the item being SF=16 in the poll-down menu in the "Binary Source". When choosing a SF, the number of bit in time slot can be computed as $3840000(chips/sec) \times 0.1667mSec \times SF$. The "Binary Source" is shown as Fig. A.3. HSDPA system provides three three modulation scheme as BPSK, QPSK, and 16-QAM. We can pick up one of them in the poll-down menu and setup the initial signal quality in the modulation block. The

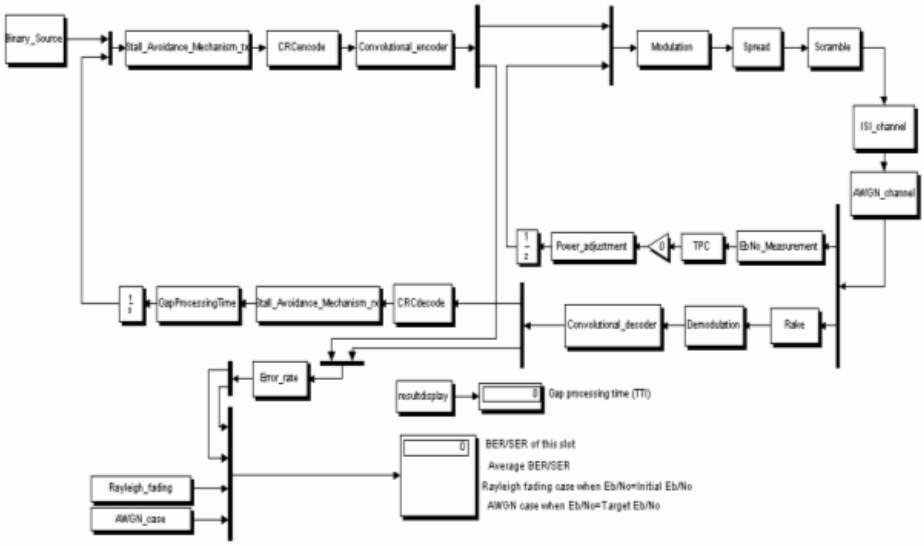


Figure A.1: The overall cross-layer simulation platform.

demodulation process is opposite to that of modulation. The spreading process is proceeded with a OVSF code. The OVSF code can be produced as the following Fig. A.4, [29]. The scrambling process doesn't further spread the bandwidth but chip-by-chip multiplying with the spreading code. The scrambling code can be produced by the following shift-register.Finally the Rake receiver can provide diversity gain and we can setup the path-diversity order in the "Rake" block. [29].

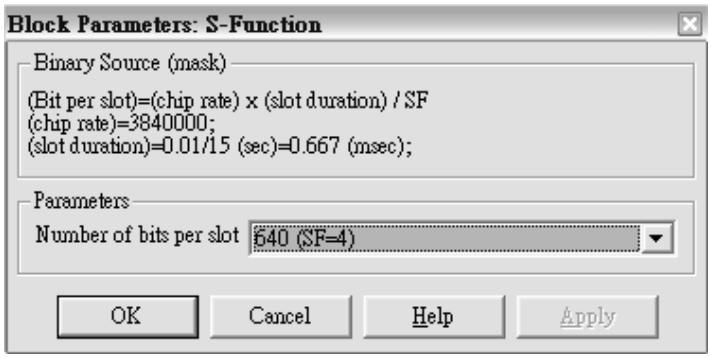


Figure A.2: The Binary Source block.

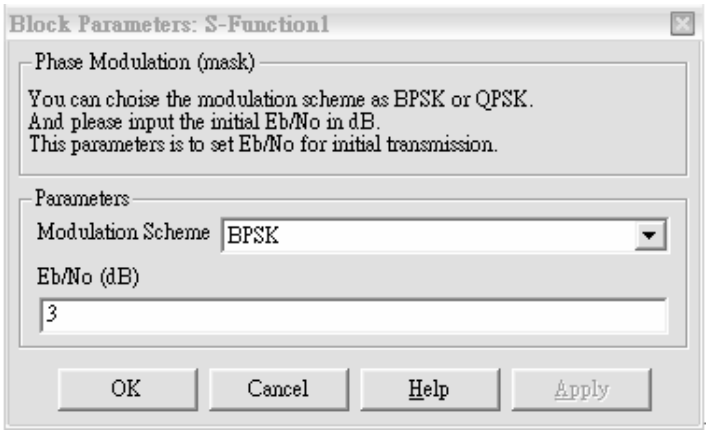


Figure A.3: The Modulation block.

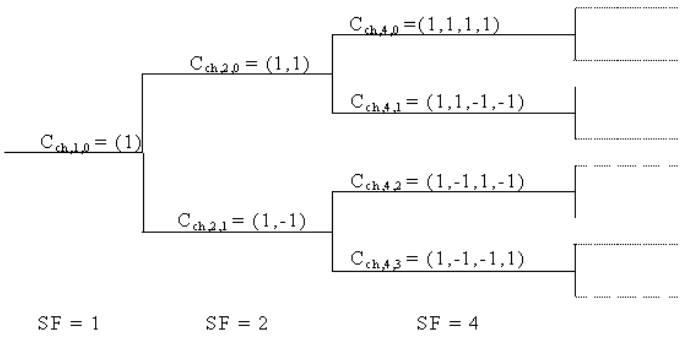


Figure A.4: The generating method of the OVSF code.

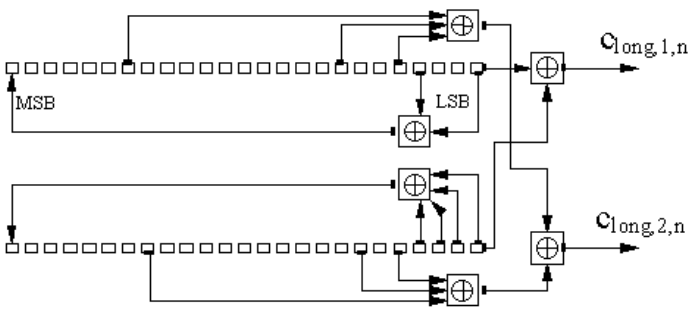


Figure A.5: The shift-register to produce the scrambling code.

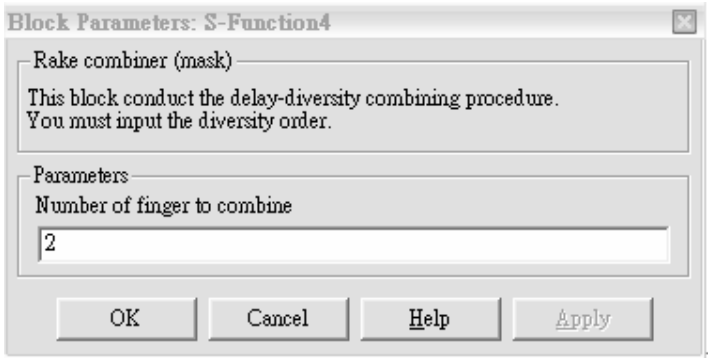


Figure A.6: The Rake receiver block.

A.2 The Channel Environment

This platform provides two environment: the AWGN and Rayleigh fading channel, as shown in Fig. A.7. The AWGN and Rayleigh fading can also be disabled by pick up the disable item. This block can produce Rayleigh fading channel and provides several power delay profile choices: no fading, urban, rural , hilly , user defined. All of these choices can be selected in the poll-down menu. The Rayleigh fading channel is produced by the Jakes model as shown in Fig. A.8 and the power delay profile in this block can be referred as Table A.10, Table A.11, and TableA.12.

A.3 The Coding Scheme

This platform both provide CRC coding with chase combining and convolutional code. The convolutional coding scheme can be selected in the poll-down menu in the "Convolutional block" and the decision length for the Viterbi decoding algorithm must be inputted.

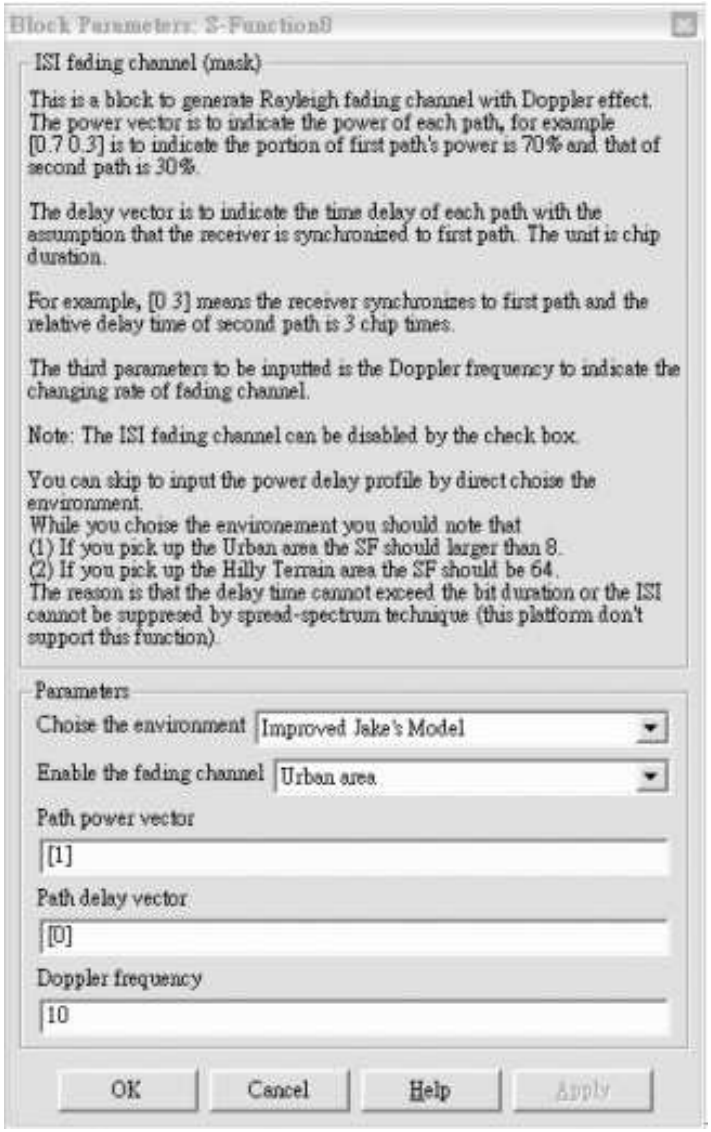


Figure A.7: ISI Channel block.

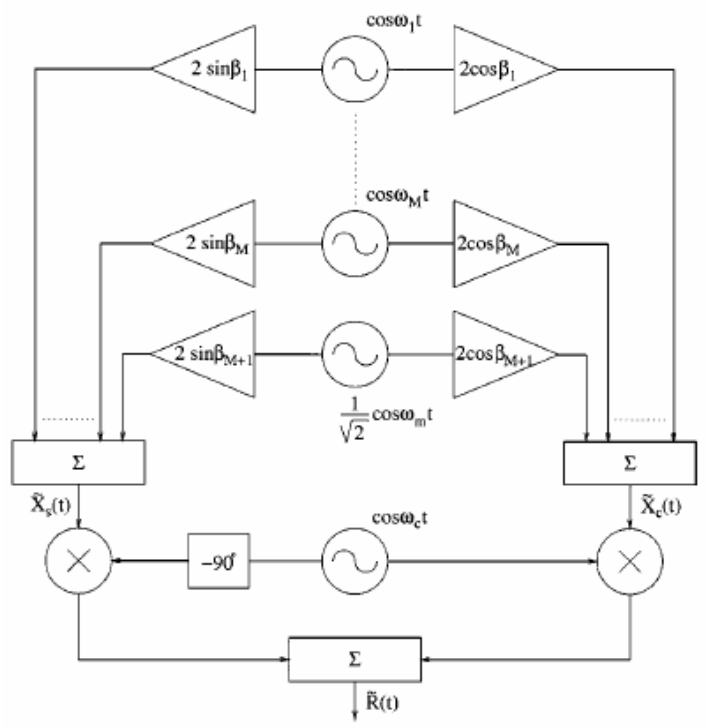


Figure A.8: The generating method of Jakes model.

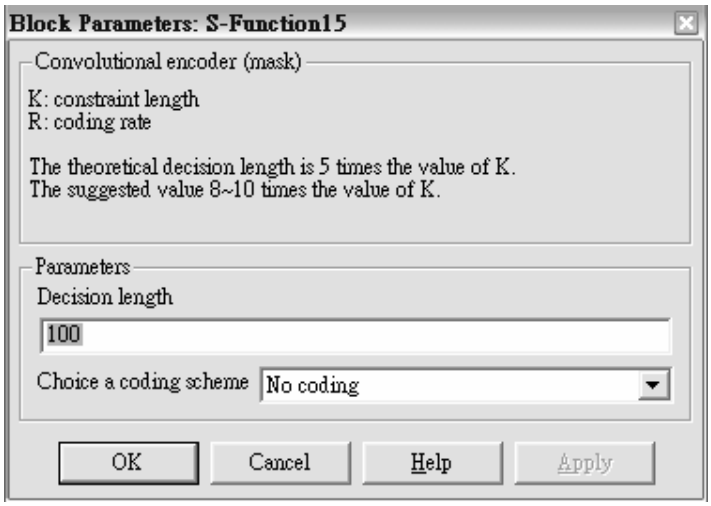


Figure A.9: The convolution encoder block.

TAP NUMBER	RELATIVE TIME (μs)	AVERAGE RELATIVE POWER (dB)	DOPPLER SPECTRUM
1	0	-5.7	Class
2	0.217	-7.6	Class
3	0.512	-10.1	Class
4	0.514	-10.2	Class
5	0.517	-10.2	Class
6	0.674	-11.5	Class
7	0.882	-13.4	Class
8	1.230	-16.3	Class
9	1.287	-16.9	Class
10	1.311	-17.1	Class
11	1.349	-17.4	Class
12	1.533	-19.0	Class
13	1.535	-19.0	Class
14	1.622	-19.8	Class
15	1.818	-21.5	Class
16	1.836	-21.6	Class
17	1.884	-22.1	Class
18	1.943	-22.6	Class
19	2.048	-23.5	Class
20	2.140	-24.3	Class

Figure A.10: The typical urban power delay profile.

TAP NUMBER	RELATIVE TIME (μs)	AVERAGE RELATIVE POWER (dB)	DOPPLER SPECTRUM
1	0	-5.2	Direct path, $f_s = 0.7 \cdot f_D$
2	0.042	-6.4	Class
3	0.101	-8.4	Class
4	0.129	-9.3	Class
5	0.149	-10.0	Class
6	0.245	-13.1	Class
7	0.312	-15.3	Class
8	0.410	-18.5	Class
9	0.469	-20.4	Class
10	0.528	-22.4	Class

Figure A.11: The rural area power delay profile.

TAP NUMBER	RELATIVE TIME (μs)	AVERAGE RELATIVE POWER (dB)	DOPPLER SPECTRUM
1	0	-3.6	Class
2	0.356	-8.9	Class
3	0.441	-10.2	Class
4	0.528	-11.5	Class
5	0.546	-11.8	Class
6	0.609	-12.7	Class
7	0.625	-13.0	Class
8	0.842	-16.2	Class
9	0.916	-17.3	Class
10	0.941	-17.7	Class
11	15.000	-17.6	Class
12	16.172	-22.7	Class
13	16.492	-24.1	Class
14	16.876	-25.8	Class
15	16.882	-25.8	Class
16	16.978	-26.2	Class
17	17.615	-29.0	Class
18	17.827	-29.9	Class
19	17.849	-30.0	Class
20	18.016	-30.7	Class

Figure A.12: The hilly mountain power delay profile.

A.4 The Stall Avoidance Mechanism

As mentioned in the chapter 3, this platform provides three stall avoidance mechanisms as timer-based, window-based and indicator-based method. Besides selecting the method for stall avoidance, we have also to setup the timer and window-size value.

A.5 The Closed-Loop Power Control

This is block also provides closed-loop power control scheme. The power control scheme are one bit up-and-down and continuous variant step methods. The related parameters, the target signal quality, have to be set up. The power control procedure can be turned off by setting the gain factor, in Fig. A.15, to be 0.

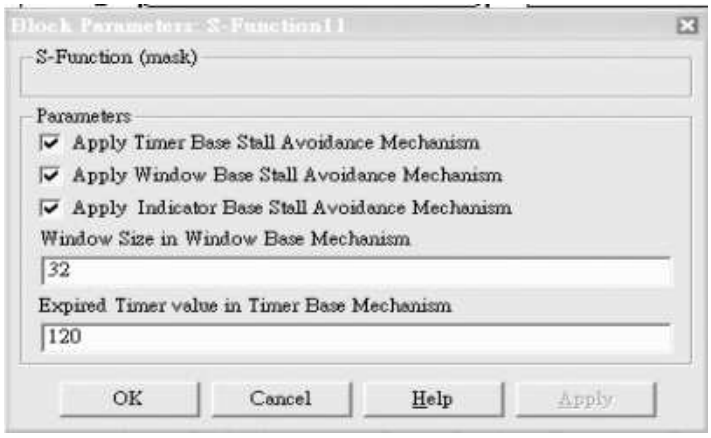


Figure A.13: The stall avoidance mechanism block.

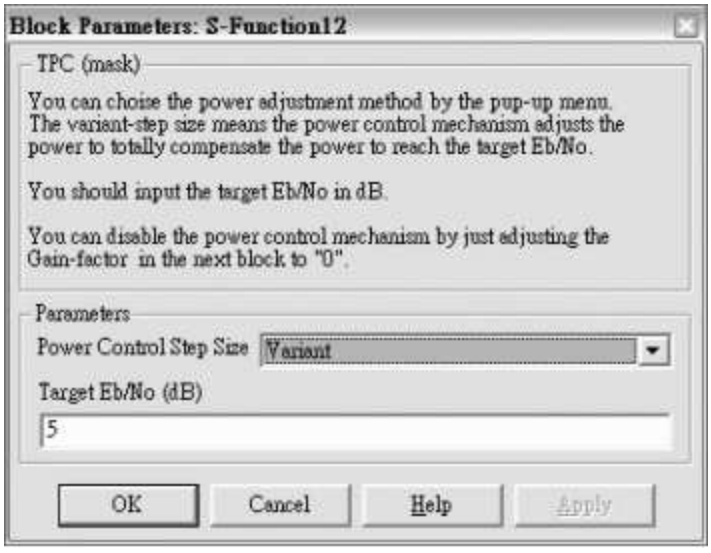


Figure A.14: The closed-loop power control scheme block.

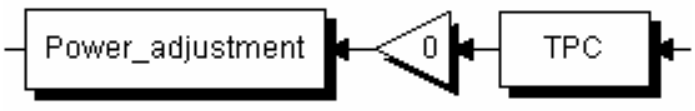


Figure A.15: The power adjustment step-size in power control procedure.

Bibliography

[1] 3GPP TR 25.950 V4.0.0, "UTRA high speed downlink packet access," March 2001.

[2] S. Parkvall, E. Dahlman, Pål Frenger, P. Beming, and M. Persson, "The high speed packet data evolution of WCDMA," *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. G27–G31, Sep. 2001.

[3] R. C. Qiu, W. Zhu, and Y.-Q. Zhang, "Third-generation and beyond (3.5G) wireless networks and its applications," *IEEE International Symposium on Circuits and Systems*, pp. I–41 – I–44, May 2002.

[4] M. Döttling, J. Michel, and B. Raaf, "Hybrid ARQ and adaptive modulation and coding schemes for high speed downlink packet access," *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1073–1077, Sep. 2002.

[5] M. Nakamura, Y. Awad, and S. Vadgama, "Adaptive control of link adaptation for high speed downlink packet access (HSDPA) in W-CDMA," *International Symposium on Wireless Personal Multimedia Communications*, pp. 382–386, Oct. 2002.

[6] R. Kwan, P. Chong, and M. Rinne, "Analysis of the adaptive modulation and coding algorithm with multicode transmission," *IEEE Vehicular Technology Conference*, pp. 2007–2011, Sep. 2002.

[7] T. E. Kolding, F. Frederiksen, and P. E. Mogensen, "Performance aspects of WCDMA systems with high speed downlink packet access (HSDPA)," *IEEE Vehicular Technology Conference*, pp. 477–481, Sep. 2002.

[8] S. Abedi and S. Vadgama, "Hybrid genetic packet scheduling and radio resource management for high speed downlink packet access," *International Symposium on Wireless Personal Multimedia Communications*, 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," *IEEE Vehicular Technology Conference*, 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," *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1462–1466, Sep. 2002.

[11] Q. Zhang and H.-J. Su, "Methods for preventing protocol stalling in UMTS radio link control," *IEEE International Conference on Communications*, 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," *IEICE Transcation on Communication*, 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," *IEEE Vehicular Technology Conference*, pp. 818–822, April 2003.

[14] A. Hottinen, J. Vesma, O. Tirkkonen, and N. Nefedov, "High bit rates for 3G and beyond using MIMO channels," *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 854–858, Sep. 2002.

[15] P. Lin and Y. B. Lin and I. Chlamtac, "Overflow control for UMTS high-speed downlink packet access," *IEEE Transaction on Wireless Communications*, vol. 3, no. 2, pp. 524–532, March 2004.

[16] 3GPP TSG-RAN WG2 R2-020945, "ACK/NACK power offsets in case of realistic channel estimation," May 2002.

[17] J. Zhang, W. Cao, M. Peng, and W. Wang, "Investigation of HYBRID ARQ performance for TDD CDMA HSDPA," *IEEE Vehicular Technology Conference*, pp. 2721–2724, April 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," *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1083–1087, Sep. 2002.

[19] R. Love, B. Classon, A. Ghosh, and M. Cudak, "Incremental redundancy for evolutions of 3G CDMA systems," *IEEE Vehicular Technology Conference*, 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," *IEEE Vehicular Technology Conference*, 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," *IEEE Vehicular Technology Conference*, pp. 1829–1833, Oct. 2001.

[22] 3GPP TS 25.308 V5.2.0, "High speed downlink packet access (HSDPA) overall description," March 2002.

[23] 3GPP TS 25.321 v5.0.0, "Medium access control (MAC) protocol specification," March 2002.

[24] 3GPP WG2-30 R2-021725, "Stall avoidance schemes in HARQ entity," June 2002.

[25] 3GPP WG2-31 R2-021974, "Stall avoidance schemes in HARQ entity," August 2002.

[26] 3GPP TSG-RAN WG2 R2-A010016, "Dual-channel stop-and-wait HARQ," January 2001.

[27] 3GPP TSG-RAN WG2 R2-021590, "Enhancements to stall avoidance mechanism," June 2002.

[28] T. S. Rappaport, W*ireless* C*ommunications,* P*rinciples and* P*ractice*, 1st ed. Prentice Hall Inc., July 1999.

[29] 3GPP TS 25.213 V4.3.0, "Spreading and modulation," June 2002.