

# 行政院國家科學委員會專題研究計畫 成果報告

## 子計畫二：無線封包網路之資源管理技術(2/2)

計畫類別：整合型計畫

計畫編號：NSC93-2219-E-009-012-

執行期間：93年08月01日至94年07月31日

執行單位：國立交通大學電信工程學系(所)

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 94 年 10 月 31 日

行政院國家科學委員會補助專題研究計畫  成果報告  
 期中進度報告

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：NSC93-2219-E009-012

執行期間：93 年 8 月 1 日至 94 年 7 月 31 日

計畫主持人：王蒞君 教授

成果報告類型(依經費核定清單規定繳交)： 精簡報告  完整報告

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

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、  
列管計畫及下列情形者外，得立即公開查詢

涉及專利或其他智慧財產權， 一年 二年後可公開查詢

執行單位：國立交通大學電信工程研究所

中 華 民 國 94 年 7 月 31 日

# 目 錄

## **Part-I:**

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

I. INTRODUCTION.....	1
II. BACKGROUND.....	3
A. Multi-Channel SAW HARQ Mechanism.....	3
B. Definition of Type-I and Type-II Gaps.....	4
III. THE STALL AVOIDANCE SCHEMES.....	5
A. Timer-Based Scheme.....	5
B. Window-Based Scheme.....	6
C. Indicator-Based Scheme.....	7
IV. PERFORMANCE MEASURE AND SYSTEM ASSUMPTIONS..	8
A. Gap-processing time.....	8
B. Assumptions.....	9
V. ANALYSIS OF TIMER-BASED STALL AVOIDANCE SCHEME.....	10
VI. ANALYSIS OF WINDOW-BASED STALL AVOIDANCE SCHEME.....	12
VII. ANALYSIS OF INDICATOR-BASED STALL AVOIDANCE SCHEME.....	15
VIII. NUMERICAL RESULTS AND DISCUSSIONS.....	20
A. Average gap-processing time of the Timer-Based Scheme.....	21
B. Average gap-processing time of the Window-Based Scheme.....	23
C. Average gap-processing time of the Indicator-Based Scheme.....	27
D. Probability mass function of the gap-processing time.....	28
IX. CONCLUSIONS.....	29
References.....	29

## **Part-II:**

### A TCP-Physical Cross-Layer Congestion Control Mechanism for the Multirate WCDMA system Using Explicit Rate Change Notification

I. INTRODUCTION.....	32
II. BACKGROUND ON TCP BEHAVIOR.....	33
III. MOTIVATION AND PROBLEM FORMULATION.....	35
A. The Relations of Queue Size, Throughput, and Delay.....	35
B. Control of the Queue Size.....	37
IV. THE PROPOSED ERCN MECHANISM.....	41
V. NUMERICAL RESULTS.....	42
A. The Simulation Model.....	42
B. The Accuracy Validation of the Analysis.....	43
C. The Comparison of ECN and ERCN.....	46
VI. CONCLUSIONS.....	48
References.....	48

### **Part-III:**

## **Comparisons of Link Adaptation Based Scheduling Algorithms for the WCDMA System with High Speed Downlink Packet Access**

I. INTRODUCTION.....	50
II. HIGH SPEED DOWNLINK PACKET ACCESS.....	53
A. Adaptive Modulation and Coding.....	54
B. Fair Scheduling.....	54
C. Multi-Code Assignment.....	54
III. CURRENT LINK ADAPTATION BASED SCHEDULING ALGORITHMS.....	55
A. Maximum C/I Scheduler.....	56
B. Fair Time Scheduler.....	56
C. Proportional Fair Scheduler.....	56
D. Exponential Rule Scheduler.....	57
IV. PROPOSED QUEUE-BASED EXPONENTIAL RULE SCHEDULER.....	58
A. Fairness Definition.....	58
B. Queue-Based Exponential Rule Scheduler.....	58
C. Time-Multiplexing Fashion v.s. Code-Multiplexing Fashion.....	59
V. SIMULATION RESULTS.....	59
A. Simulation Model.....	60
B. Single-Code Operation in the Time-Multiplexing Fashion.....	60
C. Multi-Code Operation in the Time-Multiplexing Fashion.....	62

D. Multi-Code Operation in the Code-Multiplexing Fashion.....	63
VI. CONCLUSION.....	65
References.....	66
Summary.....	71
Publications.....	73

## 摘要

這個計劃的主體主要分為三個部份(一): 寬頻分碼多重存取(WCDMA)系統中高速向下連結封包存取(HSDPA)的停滯防制(stall avoidance)機制之性能比較;(二):多速率寬頻分碼多重存取系統中使用速率變換於傳輸控制協定層(TCP)及實體層之擁塞控制機制 (三):HSDPA的封包排程(scheduling)技術之研究。

第一部份, 我們使用分析的方法比較了WCDMA中HSDPA的三種stall avoidance的方法: 計時器基礎、視窗基礎、以及指示器基礎的方法。為了達到比較的目的, 我們首先定義了一個新的評量標準稱為間空處理時間。再者, 分別推導了三種stall avoidance方法的間空處理時間的公式。藉由公式分析及模擬結果, 我們發現指示器基礎的stall avoidance的表現優於計時器基礎以及視窗基礎的方法。此外, 我們所推得的分析公式也有助於決定在停止與等待的混合自動重傳要求機制中的平行傳輸通道的數量。同時, 也可以利用此公式配合允許進入系統的用戶數。

第二部份, 我們提出了一個明確速率變換警示(ERCN)機制以改善WCDMA系統的TCP表現。我們首先推導了TCP層的最大暫存器容量與實體層之間的關係, 並且我們得到了平均流量的表示法。再這個表示法中包含了幾個變數分別是, 暫存大小, 暫時延誤及通道容量。我們所推導的公式有助於隨機變換速率的無線電系統中TCP及實體層的跨層級系統設計。

第三部份討論了高速向下連結中在考慮通道影響、延遲時間以及公平性的情況下最佳的封包排程技術。我們利用一個公平性指標評價了最大信號干擾比(maximum C/I) 排程法、知更鳥式循環(Round Robin)排程法、比例式公平(proportional fair)排程法以及指數型法則(exponential rule)排程。而後我們發現在這樣的公平性指標的定義下, 上述的方法無法達到有效的公平性原則。因此我們提出了序列式指數型法則(queue-based exponential rule)排程法, 使系統能夠同時達到公平性, 高流量以及低延遲的指標。

## Abstract

This report consists of three parts (1) gap-processing time analysis of stall avoidance schemes for high speed downlink packet access with parallel hybrid automatic repeat request (HARQ) mechanisms; (2) a transmission control protocol(TCP)-physical cross-layer congestion control mechanism for the multirate wideband code division multiple access (WCDMA) system using explicit rate change notification; (3) packet scheduling for the WCDMA system with HSDPA.

In the first part, we present an analytical approach to compare three stall avoidance schemes: the timer-based, the window-based, and the indicator-based schemes for the HSDPA in WCDMA. To this end, we first propose a new performance metric -- gap-processing time. Second, we derive the closed-form expressions for the average gap-processing time of these three stall avoidance schemes. 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 HARQ mechanisms. We also show that the analytical formulas can be used to design the number of acceptable users for an admission control policy subject to the gap-processing time constraint.

In the second part, we propose an explicit rate change notification (ERCN) mechanism to improve the TCP performance in the WCDMA system. We first derive the relation between the bandwidth in the physical layer and the maximum queue size in the TCP layer, and then obtain a closed-form expression for the average throughput in terms of average queue size, queue delay and channel capacity. We validate results by simulations. The analytical formula can facilitate the design of a cross-layer TCP-physical congestion control mechanism for the rate adaptive wireless system.

In the third part, we discuss the scheduling algorithm with the concern of channel condition, delay, and fairness for HSDPA in WCDMA systems. By a definition of fairness measure, we compare the fairness performance between maximum C/I, round robin, proportional fair, and exponential rule methods. We find that these four algorithms can not achieve fairness. Under this motivation, we propose a scheduling method called the queue-based exponential rule to reach the goal of fairness, high throughput, and low delay at the same time.

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

Li-Chun Wang<sup>1</sup> and Chih-Wen Chang

Department of Communication Engineering

National Chiao Tung University, Taiwan

Email : lichun@cc.nctu.edu.tw

## Abstract

The parallel multi-channel 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*, resulted 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 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 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 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 window size in the RLC retransmission mechanism.

<sup>1</sup>The contact author.

<sup>2</sup>This work was supported jointly by the National Science Council R.O.C under the contract 93-2219-E-009-012, 92-2219-E-009-026, EX-91-E-FA06-4-4, and the ASUSTek Computer Inc. under the contract 91C189.

<sup>3</sup>Part of this work appeared in IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, vol. 3, pp. 2431-2436, Sep. 2003.



## I. INTRODUCTION

**H**IGH 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 includes physical layer fast adaptive modulation and coding [4]–[6], fast packet scheduling in the medium access control (MAC) layer [7]–[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 multi-channel stop-and-wait (SAW) hybrid automatic repeat request (HARQ) mechanisms adopted in HSDPA. [16]–[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 non-recoverable 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 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 [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]–[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 of forwarding received packets to the upper layer as long as a gap of the received packets' sequences 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], [31].

The objective of this paper is to develop analytical methods to evaluate the performance of these three stall avoidance methods: the timer-based, the window-based, and the indicator-based schemes. To characterize the performance of the stall avoidance schemes, a new performance metric, called the gap-processing time (defined in Section III), is introduced in this paper. We derive the probability mass functions and the closed-form expressions for the average 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 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 II, we describe the stall issue in the multi-channel SAW HARQ mechanism. Section III introduces three kinds of stall avoidance schemes (the timer-based, the window-based, and the indicator-based schemes). In Section IV, we define a new performance metric — gap-processing time and describe system assumptions. Sections V ~ VII derive the average gap-processing time of the timer-based, the window-based, and the indicator-based stall avoidance schemes, respectively. Section VIII 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 IX gives our concluding remarks.

## II. BACKGROUND

### A. Multi-Channel SAW HARQ Mechanism

The multi-channel SAW HARQ is adopted in the HSDPA system [1]. The basic idea of the multi-channel SAW HARQ is to “*keep the data pipe full*”. Figure 1(a) illustrates an example of a dual-channel SAW HARQ consisting of an even transmitter and an odd transmitter [32]. As shown in Fig. 1(b), 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.

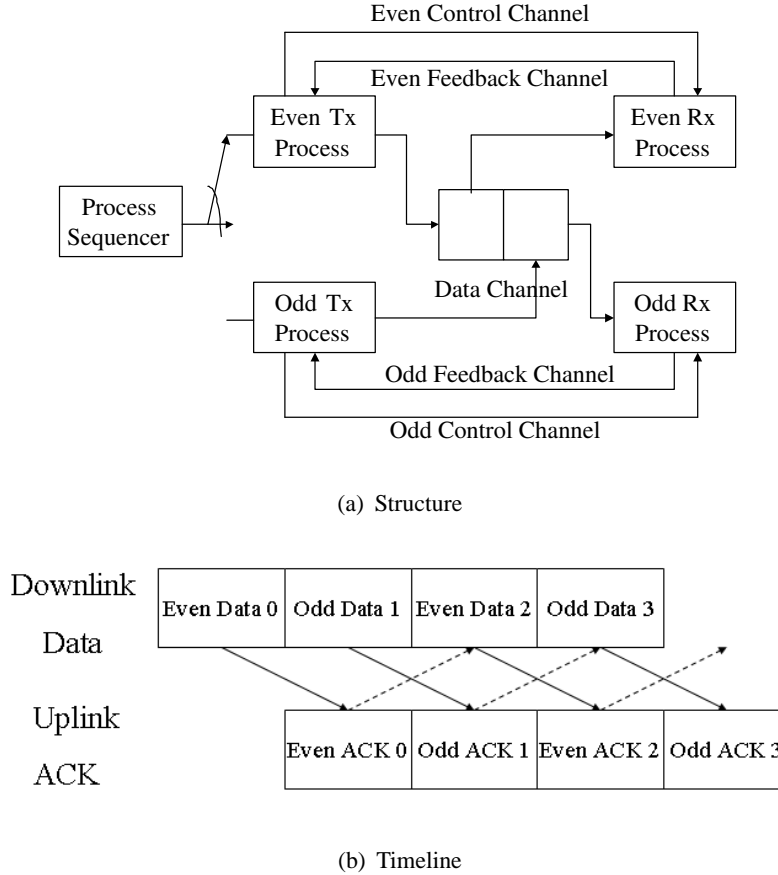


Fig. 1. The structure and timeline for the dual-channel SAW H-ARQ mechanism.

### B. 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 are 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 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].

## III. 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

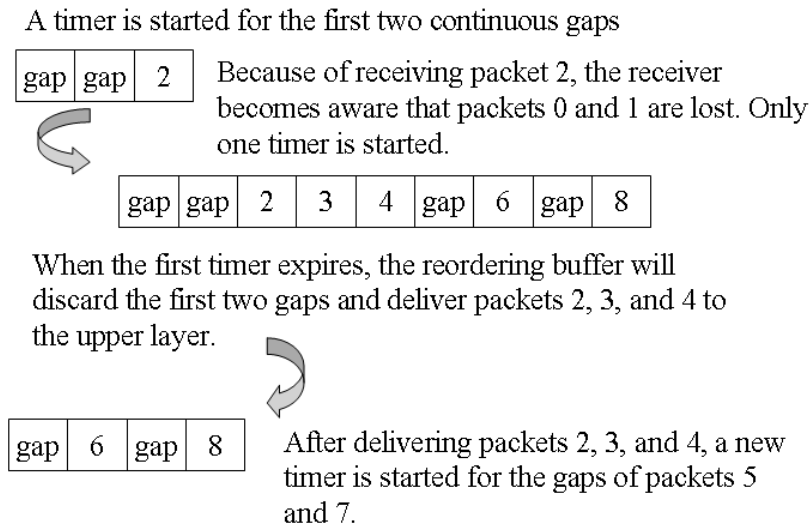


Fig. 2. An example of the timer-based stall avoidance scheme.

NACK-to-ACK error. In this section, we discuss three current stall avoidance schemes, the timer-based [27], the window-based [28], and the indicator-based schemes [29], [30].

#### A. *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 judge 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.

Figure 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-based 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 ~ 4 to the upper layer and requests for retransmitting packets 0 and 1 in the RLC layer.

### B. 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 follows. 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 a long run, a pre-determined 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 belongs 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.

Figure 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, 4 together with gaps of packet 1 and 2 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.

### C. 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 [33]. 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

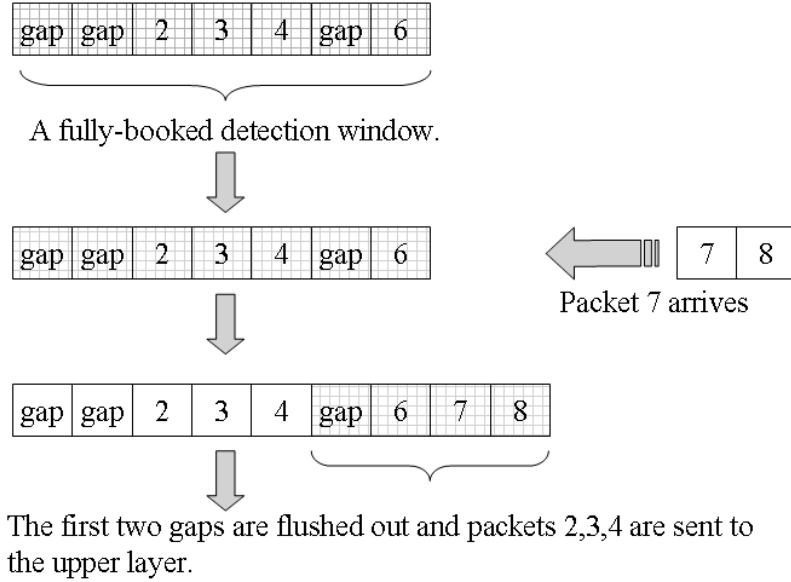


Fig. 3. An example of the window-based stall avoidance scheme with the detection window size equal to seven.

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 I 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  $TSN = 0 \sim 4$  from the RLC layer. The functions of the indicator-based stall avoidance scheme associated with Table I 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 ~ 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 ~ 2. At this moment, processes 1, 2, and 3 assumes that the missing packets 0 ~ 2 will be retransmitted, but are unsure which packet will be received. The stall avoidance scheme starts monitoring the status of processes 1, 2, and 3 to check whether these gaps belong to Type-II gaps.

TABLE I

AN EXAMPLE OF THE STATUS 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)	

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 status 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.

#### IV. PERFORMANCE MEASURE AND SYSTEM ASSUMPTIONS

##### A. 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. A short gap-processing time means the RLC retransmission can be initiated earlier, which is especially critical for the delay sensitive services. Referring to the analytical model in [34], 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 cover the effect of NACK-to-ACK error. To incorporate the effect of NACK-to-ACK error into packet delivery delay,  $T_d$  should 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. Consequently, the stall problem may seriously degrade the fluency of packet delivery. For the probability of a NACK-to-ACK error equal to 0.01, packet error rate of 0.3, and transmission time interval of 2 milli-seconds, a five seconds transmission of delay sensitive service may experience  $5/0.002 \times 0.3 \times 0.01 = 7.5$  times of the stall problem. If the stall problem can not be solved, the procedure of forwarding packet to the upper layer will be held up forever. As a result, the data delivery delay can be extremely long and the goodput can be seriously decreased. Therefore, how to solve the stall problem become an important issue, especially for the delay sensitive services. Furthermore, if the gap-processing time is longer than the RLC timeout, the excess gap-processing time may induce some additional RLC retransmission causing even longer packet delivery delay. In this paper, we assume that the RLC timeout is longer than the gap-processing time.

The longer period of gap-processing time also leads to more packets accumulated in the MAC layer. Therefore, 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-sized 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 *GPT* of 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 ( $P_{N \rightarrow A}$ ), and the MAC layer parameters, e.g. the size of the reordering buffer and the number of processes in the parallel SAW HARQ mechanisms.

### B. Assumptions

Being a function of both physical layer and MAC layer parameters, gap-processing time is difficult to be computed 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 modelled 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.



## V. 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  $P_{N \rightarrow A}$  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

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

Let  $D$  be the expiry time of the timer normalized to the transmission time interval (TTI). In terms of  $P_G$  and  $P_s$  and  $D$ , the average gap-processing time for the timer-based stall avoidance scheme (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  $\ell$  Type-II gaps (denoted by  $\overline{GPT}(\ell)$ ) is

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

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

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

(I) No new Type-II gap occurs in a period of  $D$ ,  $\ell = 0$ :

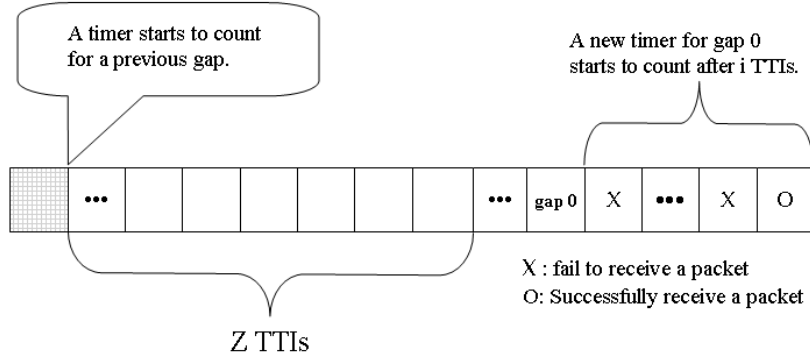
Referring to Figure 4(a), let gap 0 occur after the previous timer expires. Suppose that  $i$  TTIs later a packet with higher TSN than gap 0's is successfully received. A new timer is then initiated for gap 0. For this case, the gap-processing time of gap 0 is  $i$  plus the expiration time of the timer  $D$ . Because the probability of not having new Type-II gap within  $D$  TTIs is  $(1 - P_G)^D$  and the average time to start the timer for gap 0 is  $\sum_{i=1}^{\infty} i(1 - P_s)^{i-1} P_s$  TTIs, the average gap-processing time in the case of  $\ell = 0$  can be computed as follows:

$$\overline{GPT}(0) = \left[ D + \sum_{i=1}^{\infty} i(1 - P_s)^{i-1} P_s \right] (1 - P_G)^D . \quad (6)$$

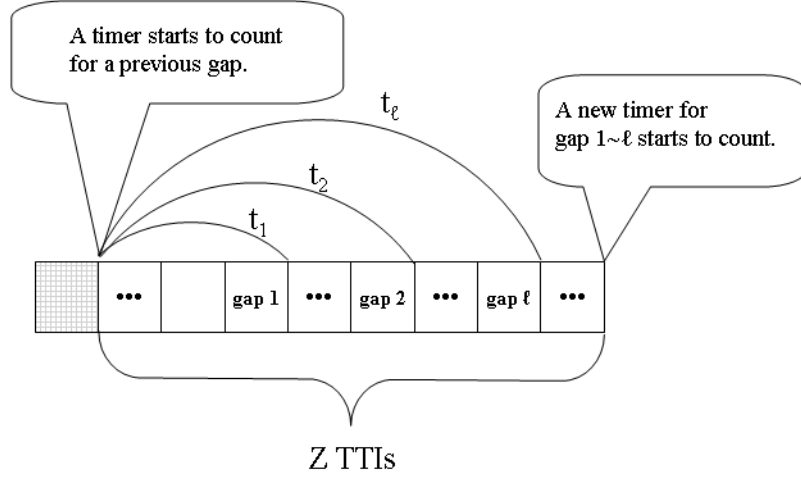
Thus, the first part of Proposition 1 is proved.

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

Assume that  $\ell$  new Type-II gaps occur before the timer expires, as shown in Fig. 4(b). For gap  $j$ , ( $j = 1, \dots, \ell$ ), 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  $j$ . The total gap-processing time for gap



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



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

Fig. 4. Two scenarios for the timer-based stall avoidance scheme to remove a Type-II gap.

$j$  is equal to  $(2D - t_j)$  in this case. As a consequence, given  $(t_1, \dots, t_\ell)$  of  $\ell$  Type-II gaps, the average gap-processing time is equal to

$$\overline{Gap}(\ell | t_1, \dots, t_\ell) = \sum_{j=1}^{\ell} \frac{2D - t_j}{\ell} . \quad (7)$$

The probability of having  $\ell$  Type-II gaps conditioned on known  $t_1, \dots, t_\ell$  can be computed by

$$\text{Prob}\{\text{having } \ell \text{ Type-II gaps} \mid t_1, \dots, t_\ell\} = P_G^\ell (1 - P_G)^{D-\ell} . \quad (8)$$

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

$$\overline{Gap}(\ell) = P_G^\ell (1 - P_G)^{D-\ell} \sum_{t_1=1}^{D-\ell+1} \sum_{t_2=t_1+1}^{D-\ell+2} \dots \sum_{t_\ell=t_{\ell-1}+1}^D \sum_{j=1}^{\ell} \frac{2D - t_j}{\ell} \quad (9)$$

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

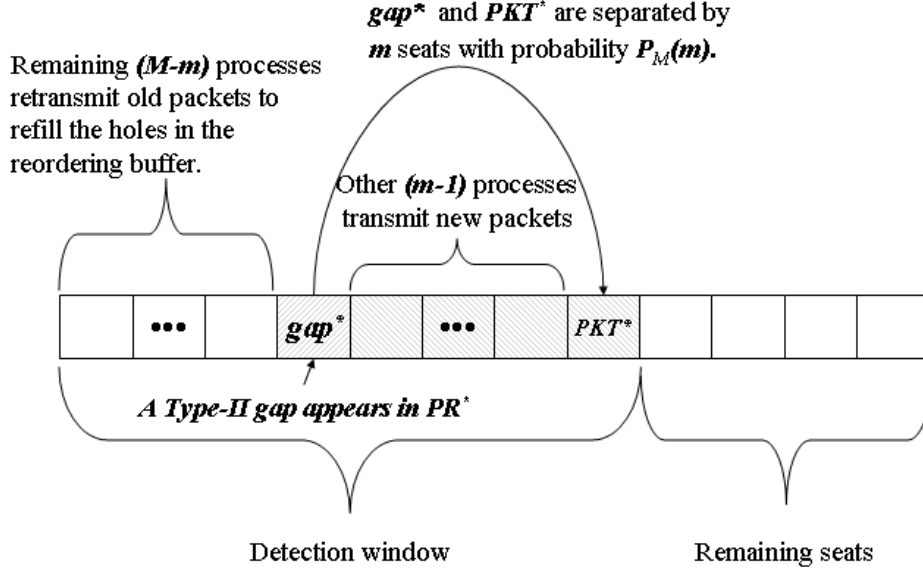


Fig. 5. An illustrative example of the seat allocation in a detection window for the window-based stall avoidance scheme.

## VI. ANALYSIS OF WINDOW-BASED STALL AVOIDANCE SCHEME

In this section, we first concisely derive the closed-form expression of the average gap-processing time for the window-based stall avoidance scheme in Proposition 2. Then, due to complexity, we separately derive its probability mass function (*pmf*) of the gap-processing time.

### A. Average Gap-processing time

**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  $P_{old}$  the probability of receiving a new packet and that of receiving a retransmitted old packet, respectively. From the definitions of  $P_s$  and  $P_{N \rightarrow A}$  in Proposition 1,  $P_{new}$  and  $P_{old}$  can be expressed as*

$$P_{new} = P_s + (1 - P_s)P_{N \rightarrow A} , \quad (10)$$

and

$$P_{old} = (1 - P_s)(1 - P_{N \rightarrow A}) . \quad (11)$$

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

$$\overline{GPT}_{window} = M + \sum_{m=1}^M \left[ (W - m) \sum_{n=1}^{\infty} n P_{new}^n P_{old}^{n-1} \right] \binom{M-1}{m-1} P_{new}^{m-1} P_{old}^{M-m} . \quad (12)$$

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

$$\overline{GPT}_{window} = \text{cycle duration} + \text{residual detection time} , \quad (13)$$

where the cycle duration is defined in Section III(C) 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 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  $i$ -th 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  $i$ -th 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 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.

- a)  $W > M$ : Now let's 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

$$P_M(m) = \binom{M-1}{m-1} P_{new}^{m-1} P_{old}^{M-m} , \quad (14)$$

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. 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  $T_o$ ) can be computed by

$$T_o = \sum_{n=1}^{\infty} n P_{new} P_{old}^{n-1} . \quad (15)$$

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 seat 1 is a Type-II gap and start the process of forwarding the received packets with type II gap to the upper later. From (14) and (15) and the above discussion, the total residual detection time for that case of  $M > M$  in addition to one cycle duration is calculated as

$$\begin{aligned} \text{residual detection time} &= \sum_{m=1}^M [(W - m)T_o] P_M(m) \\ &= \sum_{m=1}^M \left[ (W - m) \sum_{n=1}^{\infty} n P_{new} P_{old}^{n-1} \right] \binom{M-1}{m-1} P_{new}^{m-1} P_{old}^{M-m} . \end{aligned} \quad (16)$$

- 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. Following the same approach of obtaining (14) 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  $W > M$  by setting  $W = M$  in (16).

Adding the residual detection time (16) to the cycle duration ( $M$  TTIs), we obtain the average gap-processing time of the window-based stall avoidance scheme as shown in (12). ■

### B. Probability Mass Function of Gap-processing time

In this subsection, owing to the complexity, we separately derive the *pmf* of the gap-processing time for the window-based scheme. Assume that  $PKT^*$  is positioned at the  $(m+1)$ -th seat with probability  $P_M(m)$  in (14). Following from the above subsection, we know that the Type-II gap  $gap^*$  can be detected when the remaining  $(M - m - 1)$  plus one seats in the ordering buffer are occupied. Then, the gap-processing time for  $gap^*$  will be  $M + (W - m)$  TTIs when  $W - m$  consecutive new packets arrive the receiver successfully with conditional probability  $P_{new}^{W-m}$ , where  $M$  is the cycle duration. 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 belong to 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 gap-processing time equal to  $n + M$  (denoted by  $P[GPT = n + M]$ ) can be expressed as

$$\begin{aligned} P[GPT = n + M] &= P[GPT = n + M | PKT^* \text{ at } (m + 1)\text{-th seat}] P_M(m) \\ &= \binom{n-1}{W-m-1} P_{new}^{W-m} P_{old}^{n-(W-m)} \binom{M-1}{m-1} P_{new}^{m-1} P_{old}^{M-m} \\ &= \binom{n-1}{W-m-1} \binom{M-1}{m-1} P_{new}^{W-1} P_{old}^{n+M-W}. \end{aligned} \quad (17)$$

Note that (17) 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 after  $n + 1$  transmissions, where  $n \geq 0$ . Then, we can have

$$\sum_{m=1}^M \sum_{n=W-m}^L P[GPT = n + M] = 1, \quad (18)$$

where  $L$  is the required TTIs to detect a Type-II gap. The average gap-processing time can also be calculated by  $\sum_{m=1}^M \sum_{n=W-m}^L (n + M) P[GPT = n + M]$ , which is equal to the result obtained from (12).

## VII. 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 for the indicator-based scheme can be calculated in terms of parameters  $M$ ,  $P_s$ ,  $P_{new}$ , and  $P_{old}$  as follows:*

$$\begin{aligned} \overline{GPT}_{indicator} &= M(P_{new} + P_{old}P_s)^{M-1} + \\ &\sum_{k=1}^C \sum_{m=2}^M (m-1 + kM) P(x_k = S_1) \left[ \sum_{i=0}^k P(x_i = S_1) \right]^{m-2} \left[ \sum_{j=0}^{k-1} P(x_j = S_1) \right]^{M-m}, \end{aligned} \quad (19)$$

where

$$P(x_0 = S_1) = P_{new} + P_{old}P_s \quad (20)$$

and

$$P(x_\ell = S_1) = P_{old}^\ell(1 - P_s)P_{new} \quad \text{for } \ell \geq 1. \quad (21)$$

$C$  is the required cycles of involving all the processes in the  $M$ -channel SAW HARQ mechanism 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 are described by the following two events:

$$\mathbf{A} \triangleq \{NDI_{(rec)} = NEW\} \quad ; \quad (22)$$

$$\mathbf{B} \triangleq \{\{TSN_{(rec)} \neq TSN^*\} \cap \{NDI_{(rec)} = OLD\}\} \quad , \quad (23)$$

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 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  $x_\ell$  the state variable in the  $\ell$ -th cycle. If a process in the retransmission state in the  $(\ell - 1)$ -th cycle, the receiving process will issue a NACK signal for requesting a retransmission in the  $\ell$ -th cycle. However, if the NACK signal is changed to an ACK signal, a new packet will be transmitted in the  $\ell$ -th cycle instead. Based on the definition of event  $\mathbf{A}$  in (22), we have

$$P(\mathbf{A}|x_{\ell-1} = S_0) = P_{N \rightarrow A} \quad , \quad (24)$$

where  $P_{N \rightarrow A}$  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  $\ell$ -th cycle. Note that probability of the NACK signal arriving the transmitter correctly is  $(1 - P_{N \rightarrow A})$  and the probability of successfully retransmitting packet to the receiver is  $P_s$ . According to the definition of (23), the occurrence probability of event **B** is equal to

$$P(\mathbf{B}|x_{\ell-1} = S_0) = (1 - P_{N \rightarrow A})P_s . \quad (25)$$

Because events **A** and **B** are mutually exclusive and from (24) and (25), we can obtain

$$\begin{aligned} P(x_\ell = S_0|x_{\ell-1} = S_0) &= P(\overline{\{\mathbf{A} \cup \mathbf{B}\}}|x_{\ell-1} = S_0) \\ &= 1 - [P(\mathbf{A}|x_{\ell-1} = S_0) + P(\mathbf{B}|x_{\ell-1} = S_0)] \\ &= 1 - [P_{N \rightarrow A} + (1 - P_{N \rightarrow A})P_s] \\ &= (1 - P_{N \rightarrow A})(1 - P_s) \\ &= P_{old} . \end{aligned} \quad (26)$$

Recall that in (11), we define  $P_{old} = (1 - P_{N \rightarrow A})(1 - P_s)$ . According to the definition of  $S_0$ , when a process receives a retransmitted old packet which fails the CRC test, it enters state  $S_0$ . Thus, the initial probability of a process at  $S_0$  can be expressed as

$$P(x_0 = S_0) = P_{old}(1 - P_s) . \quad (27)$$

- 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 and (24) and (25), the probability of a process in the stop state  $S_1$  in the  $\ell$ -th cycle can be expressed as

$$\begin{aligned} P(x_\ell = S_1|x_{\ell-1} = S_0) &= P(\mathbf{A} \cup \mathbf{B}|x_{\ell-1} = S_0) \\ &= P(\mathbf{A}|x_{\ell-1} = S_0) + P(\mathbf{B}|x_{\ell-1} = S_0) \\ &= P_{N \rightarrow A} + (1 - P_{N \rightarrow A})P_s \\ &= P_s + (1 - P_s)P_{N \rightarrow A} \\ &= P_{new} . \end{aligned} \quad (28)$$

Recall that in (10), we define  $P_{new} = P_s + (1 - P_s)P_{N \rightarrow A}$ . 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

$$P(x_0 = S_1) = P_{new} + P_{old}P_s . \quad (29)$$



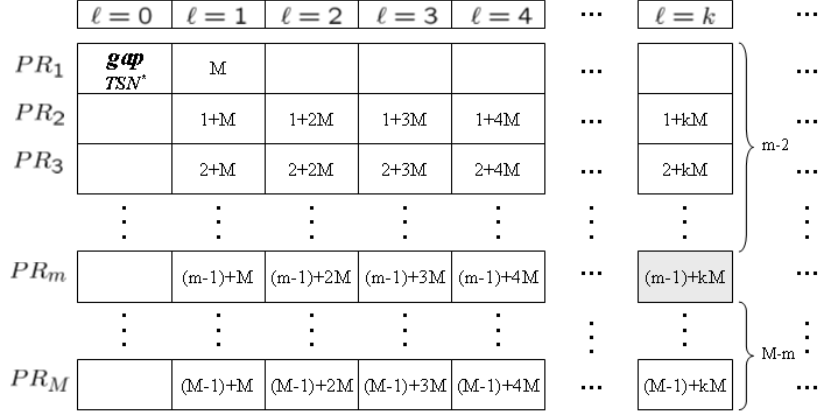


Fig. 6. An illustration for the gap-processing time of the indicator-based stall avoidance scheme.

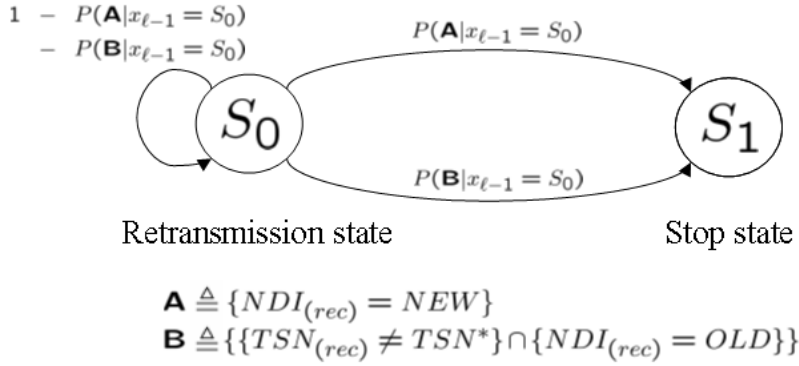


Fig. 7. The state transition diagram for the indicator-based stall avoidance scheme.

From (26) and (28), we obtain the state transition probability matrix  $\Gamma$  of the two-state Markov chain in Fig. 7 as follows:

$$\begin{aligned}
 \Gamma &\triangleq \begin{bmatrix} P(x_\ell = S_0|x_{\ell-1} = S_0) & P(x_\ell = S_1|x_{\ell-1} = S_0) \\ P(x_\ell = S_0|x_{\ell-1} = S_1) & P(x_\ell = S_1|x_{\ell-1} = S_1) \end{bmatrix} \\
 &= \begin{bmatrix} P_{old} & P_{new} \\ 0 & 0 \end{bmatrix}. \tag{30}
 \end{aligned}$$

Then from (27) and (29), a state probability vector  $\mathbf{P}(\mathbf{x}_\ell) \triangleq [P(x_\ell = S_0), P(x_\ell = S_1)]$  can be obtained by

$$\begin{aligned}
 \mathbf{P}(\mathbf{x}_\ell) &= \mathbf{P}(\mathbf{x}_{\ell-1})\Gamma = \mathbf{P}(\mathbf{x}_0)\Gamma^\ell \\
 &= [P_{old}^{\ell+1}(1 - P_s), P_{old}^\ell(1 - P_s)P_{new}]. \tag{31}
 \end{aligned}$$

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

(I) 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 (29), the average gap-processing time for detecting a Type-II gap immediately after cycle 0 can be expressed as

$$\begin{aligned} \overline{GPT}_0 &= M \times [P(x_0 = S_1)]^{M-1} \\ &= M(P_{new} + P_{old}P_s)^{M-1} . \end{aligned} \quad (32)$$

(II) 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-II gap. Denote  $P_{Ind}(m, k)$  the probability the Type-II gap being detected in the  $k$ -th cycle by  $PR_m$ , where  $m \geq 2$  and  $k \geq 1$ . As shown in Fig. 6, the gap-processing time for this case is  $(m - 1 + kM)$ . To compute  $P_{Ind}(m, k)$ , three event  $E_1$ ,  $E_2$ , and  $E_3$  are defined as follows:

$$E_1 = \{PR_m \text{ is ruled out in the } k\text{-th cycle.}\} \quad (33)$$

$$E_2 = \{PR_2 \sim PR_{m-1} \text{ are ruled out within } k\text{-th cycle.}\} \quad (34)$$

$$E_3 = \{PR_{m+1} \sim PR_M \text{ are ruled out within } (k - 1)\text{-th cycle.}\} \quad (35)$$

From the state probability of the Markov chain in (31), it is obvious that

$$P(E_1) = P(x_k = S_1) . \quad (36)$$

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

$$\begin{aligned} P(E_2) &= \overbrace{P_k^{(2)} \times P_k^{(3)} \times \dots \times P_k^{(m-1)}}^{(m-2)} \\ &= \left[ \sum_{i=0}^k P(x_i = S_1) \right]^{m-2} . \end{aligned} \quad (37)$$

Similarly, for event  $E_3$  we can have

$$P(E_3) = \left[ \sum_{j=0}^{k-1} P(x_j = S_1) \right]^{M-m} . \quad (38)$$

Since events  $E_1$ ,  $E_2$ , and  $E_3$  are mutually independent, combining (36), (37), and (38), we can express the average gap-processing time for case (II) as follows:

$$\begin{aligned}
\overline{GPT}_1 &= \sum_{k=1}^C \sum_{m=2}^M (m-1+kM) P_{Ind}(m, k) \\
&= \sum_{k=1}^C \sum_{m=2}^M (m-1+kM) P(E_1) \times P(E_2) \times P(E_3) \\
&= \sum_{k=1}^C \sum_{m=2}^M (m-1+kM) P(x_k = S_1) \left[ \sum_{i=0}^k P(x_i = S_1) \right]^{m-2} \left[ \sum_{j=0}^{k-1} P(x_j = S_1) \right]^{M-m}, \tag{39}
\end{aligned}$$

where  $C$  is the required cycles of involving all the processes of an  $M$ -channel SAW HARQ mechanism 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

$$(P_{new} + P_{old}P_s)^{M-1} + \sum_{k=1}^C \sum_{m=2}^M P(x_k = S_1) \left[ \sum_{i=0}^k P(x_i = S_1) \right]^{m-2} \left[ \sum_{j=0}^{k-1} P(x_j = S_1) \right]^{M-m} = 1. \tag{40}$$

Combining (32) and (39), the average gap-processing time for the indicator-based stall avoidance scheme can be computed by

$$\begin{aligned}
\overline{GPT}_{indicator} &= \overline{GPT}_0 + \overline{GPT}_1 \\
&= M(P_{new} + P_{old}P_s)^{M-1} + \\
&\quad \sum_{k=1}^C \sum_{m=2}^M (m-1+kM) P(x_k = S_1) \left[ \sum_{i=0}^k P(x_i = S_1) \right]^{m-2} \left[ \sum_{j=0}^{k-1} P(x_j = S_1) \right]^{M-m}. \tag{41}
\end{aligned}$$

■

## VIII. NUMERICAL RESULTS AND DISCUSSIONS

In this section, by analysis and simulations, we investigate the average gap-processing time and the corresponding probability mass function 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 users for different stall avoidance schemes. For these purposes, we conduct a simulation cross physical and MAC layers under the assumptions of IV.(B). In the simulation, each packet is transmitted through the flat Rayleigh fading channel. The correctness of each received packet is checked by CRC. If the received packet passes CRC, a ACK signal will be sent through

<sup>1</sup>The  $PER$  of the 1<sup>st</sup> transmission in the fast retransmission mechanism in HSDPA can be 50% [22].

TABLE II  
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 ~ 20 dB
$1^{st}$ transmission $PER$ <sup>1</sup>	0.09 ~ 0.42

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. Then, the received packets are accumulated in the reordering buffer until that Type-II gap is detected by the stall avoidance scheme. The gap-processing time is calculated from its appearance until it is detected. To confirm the correctness of simulation results, we run 100000 packets for each simulation. The Doppler frequency of the Rayleigh fading channel is 100 Hz, which is equivalent to 54 km/hour with a carrier frequency of 2 GHz. Other simulation parameters are shown in Table II. Furthermore, in the discussion of the number of acceptable users, we assume that each user contributes equal traffic load to the system and the system capacity can support the all the requests from users. Also, all the users are 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 multiple users.

#### A. Average gap-processing time of the Timer-Based Scheme

Figure 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% smaller than the simulation value (22.3 TTIs). Regarding the 2.7% discrepancy of the analytical result, the following summarized the reason. Recall that the stall avoidance scheme will be triggered when the receiver successfully receives a packet with a TSN larger than that of a gap in the reordering buffer. However, this rule may be conflicted in the derivations of Proposition I. Referring to Fig. 4(a), a timer begins to count for gap 0 when the packet denoted by "O" reaches the receiver under the assumption that the packet "O" possesses a TSN larger than that of gap 0. This is not always the case for packet "O" carrying a larger TSN. If the TSN of packet "O" is less than that of gap 0, the receiver needs

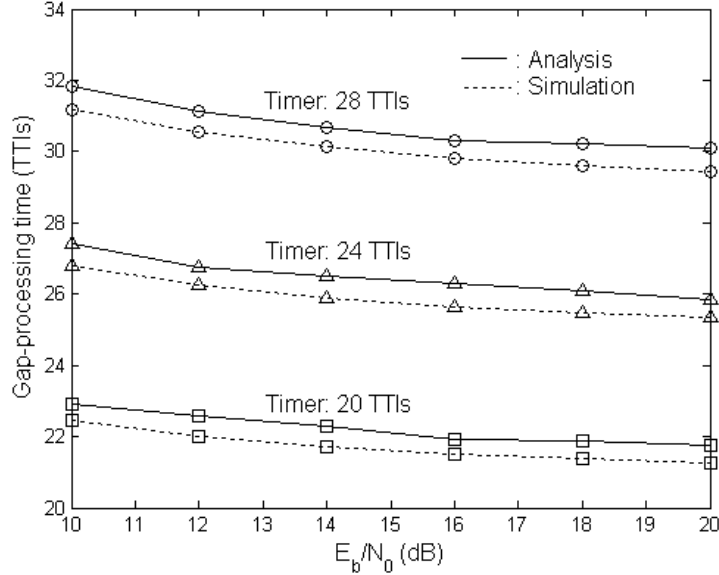


Fig. 8. The average gap-processing time of the timer-based stall avoidance scheme with different timer expiration for the 4-channel SAW HARQ mechanism in the Rayleigh fading channel with Doppler frequency of 100 Hz.

to wait for a next successful packet with larger TSN to discover gap 0. In this case, the gap-processing time will be extended. The same phenomenon happens in another case of Proposition I. As shown in Fig. 4(b), a new timer starts for gap  $1 \sim \ell$  immediately after the previous timer expires under the assumption that the packet (denoted by  $PKT^*$ ) received in the last time slot of the counting period of the last timer carries a TSN larger than that of gap  $\ell$ . However, in the following two cases, the new timer will not count for gap  $\ell$ : (1) the TSN of  $PKT^*$  is smaller than that of gap  $\ell$ ; (2)  $PKT^*$  has a TSN larger than that of gap  $\ell$  but it fails CRC. In these two situations, gap  $\ell$  will not be discovered and included in the counting period of the new timer. Consequently, the gap-processing time of gap  $\ell$  will be extended. This explains the lower values of the analytical results.

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  $\overline{GPT}_{timer} = 30.1$  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 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  $P_{N \rightarrow A}$  without time consuming simulations.

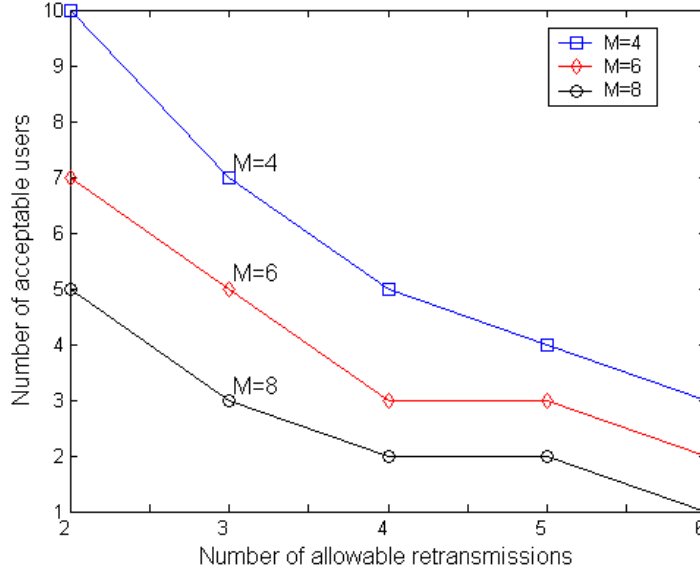


Fig. 9. The number of acceptable users of the timer-based stall avoidance scheme versus the numbers of allowable retransmissions ( $h$ ) for various number processes ( $M$ ) in the parallel SAW HARQ mechanism subject to a constraint of gap-processing time 100 TTIs.

Furthermore, the derived analytical method of computing the gap-processing time can also be applied to design the proper number of acceptable users from the admission control standpoint. Let  $T_o$  be the constraint on the maximum allowable gap-processing time. Then the acceptable users in the system can be approximated by  $\lfloor T_o / \overline{GPT}_{timer} \rfloor$ , where  $\lfloor x \rfloor$  is the function of obtaining a maximum integer less than  $x$ . According to the above guidelines, Figure 9 shows the number of acceptable 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 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 users. Similarly, as the number of parallel HARQ processes increases, the effect of increasing gap-processing time will reduce the acceptable users.

### B. Average gap-processing time of the Window-Based Scheme

Figure 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. (12)) 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-

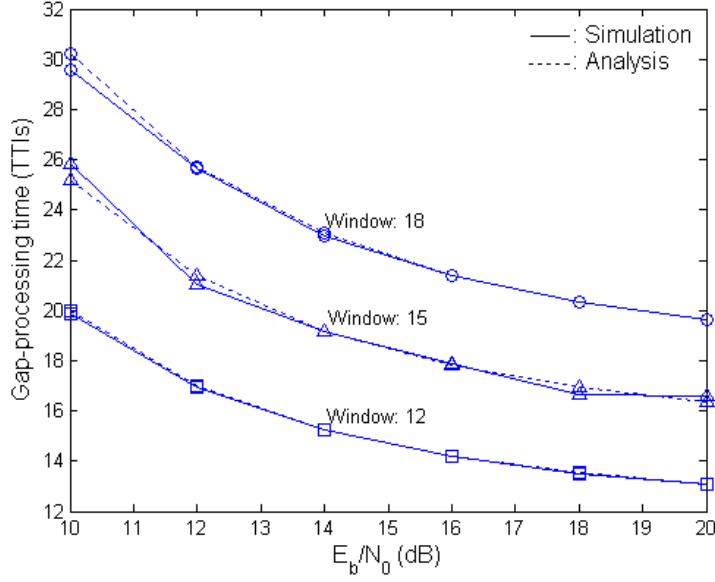


Fig. 10. The average gap-processing time of the window-based stall avoidance scheme with different window sizes for the 4-channel SAW HARQ mechanism in the Rayleigh fading channel with Doppler frequency of 100 Hz.

processing time constraint  $T_0$ . Consider a fair scheduling policy. The number of acceptable users ( $N$ ) in the system can be approximated by

$$N = \lfloor T_0 / \overline{GPT}_{window} \rfloor, \quad (42)$$

where  $\lfloor x \rfloor$  is the function of obtaining a maximum integer less than  $x$ . 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

$$n = \frac{W}{M-1} - 1. \quad (43)$$

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 3 processes transmit new packets successfully. After 4 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. Figure 11 shows the

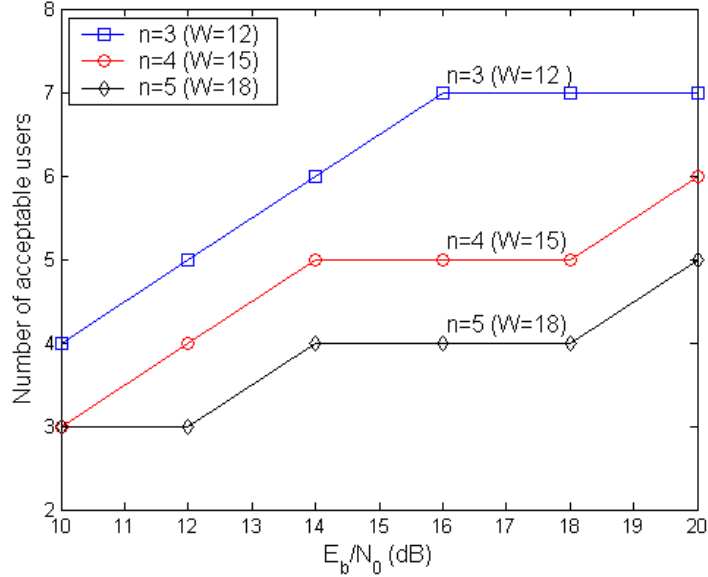


Fig. 11. The number of acceptable users of the window-based stall avoidance scheme versus  $E_b/N_0$  with various number minimum allowable retransmissions ( $n = 3, 4, 5$ ) in the 4-process SAW HARQ mechanism subject to a gap-processing time constraint of 100 TTIs.

number of acceptable 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 (42). From (43),  $n = 3, 4$ , and  $5$  correspond to  $W = 12, 15$ , and  $18$ , respectively. As shown in the figure, the acceptable users is reduced from 7 to 4 as  $n$  increases from 3 to 5 for  $16 \text{ dB} \leq E_b/N_0 \leq 18 \text{ dB}$ .

Figure 12 shows the number of acceptable 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  $n = 3$ . These curves are obtained by substituting the parameters  $M, W, P_s$ , and  $P_{N \rightarrow A}$  into (12). Note that  $PER = 1 - P_s$  and the corresponding window size  $W = 12, 20$ , and  $28$  is obtained from  $W = (n + 1)(M - 1)$  according to (43). This figure can be associated with an admission control policy subject to gap-processing time by observing  $PER$ . According to the CRC results and  $PER$ , 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 users decreases from 7 to 3 as  $M$  increases from 4 to 8. Recall  $n = \frac{W}{M-1} - 1$  in (43). For a fixed  $n$ , a larger value of  $M$  also leads to a larger value of  $W$  and longer gap-processing time. Hence, the number of acceptable users is reduced for a larger value of  $M$  to satisfy the gap-processing time requirement.



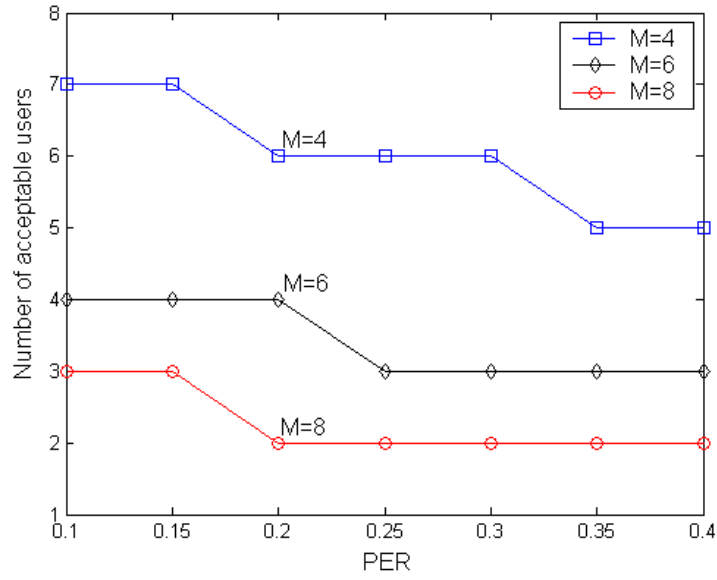


Fig. 12. The number of acceptable users of the window-based stall avoidance scheme versus  $PER$  with various number processes ( $M$ ) in the parallel SAW HARQ mechanism subject to a gap-processing time constraint of 100 TTIs. The minimum allowable retransmission ( $n$ ) is three.

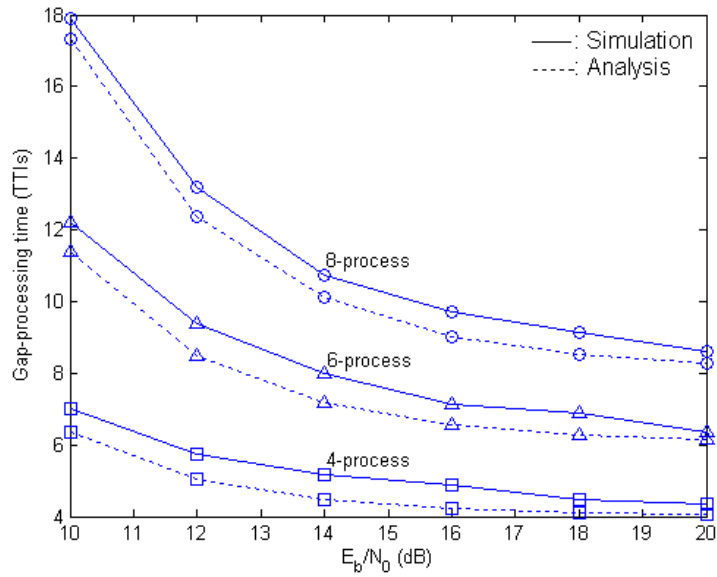


Fig. 13. Effect of the number of processes in the multi-channel SAW HARQ mechanism on the gap-processing time for the indicator-based avoidance scheme in the Rayleigh fading channel with Doppler frequency of 100 Hz .

### C. Average gap-processing time of the Indicator-Based Scheme

Figure 13 shows the analytical average gap-processing time for the indicator-based stall avoidance scheme obtained from (19) and simulations. As shown in the figure, the differences between the two are quite small. The reason for the difference is similar to that for the discrepancy of the timer-based scheme. Referring to Fig. 6, it indicates that the minimum gap-processing time for the indicator-based stall avoidance is  $M$  when processes  $PR_2 \sim PR_M$  receive packet successfully. However, if the packet in process  $PR_2$  carries a TSN smaller than  $TSN^*$  of the gap in process  $PR_1$  at  $\ell = 0$ , then the indicator-based stall avoidance scheme can not be triggered to monitor the activities of the parallel processes at  $\ell = 0$  of process  $PR_2$  even if process  $PR_2$  receives a packet successfully at  $\ell = 0$ . Then, it may take another cycle for process  $PR_2$  to enter the stop state defined in Section IV. Consequently, the gap-processing time must be longer than  $M$  in this situation. Furthermore, there could be some kind of probability for the indicator-based scheme to finish the monitoring procedures at cycle  $\ell > 1$  of process  $PR_1$ . Consider that (1) the TSNs of processes  $PR_2 \sim PR_M$  at cycle  $\ell = 0$  are all smaller than  $TSN^*$ ; (2) process  $PR_1$  receives a new packet which fails the CRC at cycle  $\ell = 1$ ; (3) processes  $PR_2 \sim PR_M$  successfully receive packets with TSNs larger than  $TSN^*$  at cycle  $\ell = 1$ . In this situation, the monitoring procedure of the indicator-based stall avoidance scheme will be finished whenever process  $PR_1$  receives a packet successfully at cycle  $\ell > 1$ . Especially note that the probability of the occurrence of the above joint conditions of (1)~(3) is larger for a smaller  $M$ . This explains the larger discrepancy of the analytical results of  $M = 4$ . The above cases are excluded from the derivations of Proposition III because it is very difficult to include the information of the TSN into derivations. Thus, actually, Proposition III provides an elegant approximation of the average gap-processing time for the indicator-based scheme.

Comparing 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 users through  $\lfloor T_o / \sqrt{GPT}_{indicator} \rfloor$  as the timer-based and the window-based schemes, where  $T_o$  is a given constrain on the gap-processing time. Figure 14 shows the number of acceptable 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 scheme, an HSDPA system can accommodate more

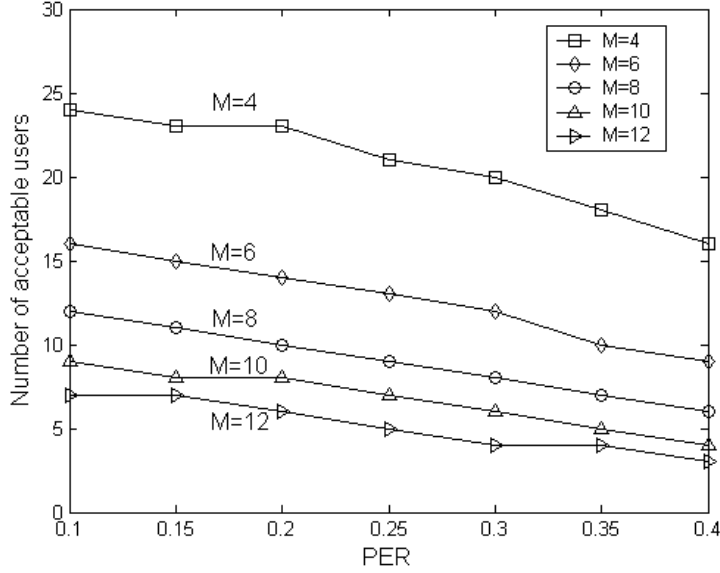


Fig. 14. The number of acceptable users of the indicator-based stall avoidance scheme versus  $PER$  with various number processes ( $M$ ) in the parallel SAW HARQ mechanism subject to a gap-processing time constraint of 100 TTIs.

users compared to the window-based scheme. For  $M = 4$  at  $PER = 0.2$ , the number of acceptable 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 users from 24 to 6 if the gap-processing time requirement is fulfilled.

#### D. Probability Mass Function of the Gap-processing Time

Figure 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. 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% and 71% 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% 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 the packet data access.

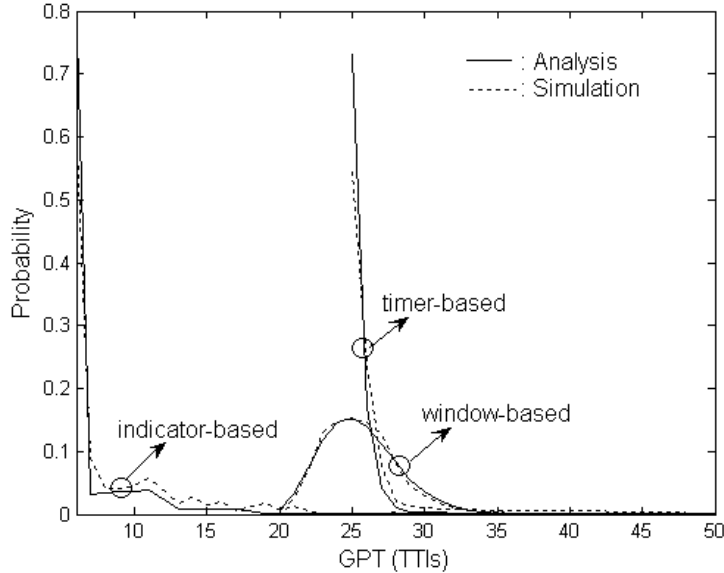


Fig. 15. The probability mass functions of the gap-processing time for the timer-based, the window-based, and the indicator-based stall avoidance schemes, where the timer's expiration  $D = 24$  and the detection window size  $W = 20$  with  $M = 6$  parallel HARQ processes at  $E_b/N_0 = 14$  dB.

## IX. 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 and the window size in the RLC layers, and the number of acceptable 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.

**Acknowledgement:** The authors wish to thank Dr. Sam Jiang for his helpful suggestions and ASUSTek Computer Inc. for its sponsorship in this project.

## REFERENCES

- [1] 3GPP TR 25.950 V4.0.0, "UTRA high speed downlink packet access," March 2001.
- [2] S. Parkvall, E. Dahlman, P. 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. 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," *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 scheduling in high-speed downlink packet access," *IEICE Transaction 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] M. Chatterjee, G. D. Mandyam, and S. K. Das, "Fast ARQ in high speed downlink packet access for WCDMA systems," *Proceedings of European Wireless*, pp. 451–457, Feb. 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, asynchronous 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] G. Manuel and M. Rinne, "Performance of the medium access control protocol for the high speed downlink packet access," *Proceedings of the IASTED International Conference Communication Systems and Networks*, pp. 42–47, Sep. 2003.
- [23] G. Manuel and M. Rinne, "Analysis of the transmission window for the delay performance of the high speed downlink packet access protocol," *International Conference on Software, Telecommunications 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 requirements 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," March 2002.
- [28] 3GPP TS 25.321 V5.0.0, "Medium access control (MAC) protocol specification," March 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," August 2002.
- [31] M. Chatterjee, G. D. Mandyam, and S. K. Das, "Fast ARQ in high speed downlink packet access for WCDMA systems," *Proceedings of European Wireless*, pp. 451–457, Feb. 2002.
- [32] 3GPP TSG-RAN WG2 R2-A010016, "Dual-channel stop-and-wait HARQ," January 2001.
- [33] 3GPP TR 25.855 V5.0.0, "High speed downlink packet access overall UTRAN description," Sep. 2001.
- [34] H. Lin and S. K. Das, "Performance study of link layer and MAC layer protocols to support TCP in 3G CDMA systems," *IEEE Trans. on mobile computing*, vol. 4, no. 5, pp. 489–502, Sept./Oct. 2005.

# A TCP-Physical Cross-Layer Congestion Control Mechanism for the Multirate WCDMA system Using Explicit Rate Change Notification

Li-Chun Wang and Ching-Hao Lee

## I. INTRODUCTION

Due to the fact that the WCDMA system adapts its transmission rate according to the radio link quality, transmitting TCP traffic in the WCDMA system becomes a difficult but challenging task [1] and [2]. To this end, a base station is usually equipped with a buffer to accommodate for the load variations of TCP traffic. The buffer management has two major considerations. One is the link utilization, and the other is the queueing delay. The former consideration prefers to a larger queue, while the latter consideration prefers to a smaller-sized queue [3], [4] and [5].

In the wireline network, the active queue management (AQM) mechanism and the random early detection (RED) are two well-know buffer management techniques for the Internet routers [6] and [7]. The basic principle of AQM and RED is to react on buffer overload before the buffer reaches its capacity limit.

However, in the wireless link, there exist different issues in buffer management for delivering TCP traffic in the WCDMA system. There issues includes rate variation, long latency [8] and [9], and per-host queueing [3]. Thus, in [3], they proposed the packet discard prevention counter (PDPC) method. In [3], the major goal was focussed on the maximization of radio link utilization, while the impact of queue delay and link delay jitter was not a key design consideration.

In this work, by analyzing the relation of bandwidth in the physical layer and the queue size in the TCP layer, we obtain a closed-form expression for the throughput in the TCP layer in terms of queue size, queue delay and channel capacity. Accordingly, we suggest a physical-TCP cross-layer buffer management technique to improve the TCP throughput and queue delay and link delay jitter.

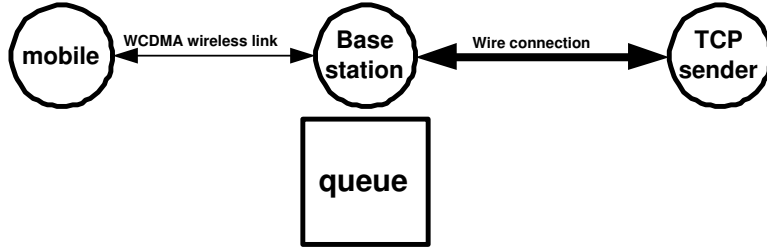


Fig. 1. The TCP flow in the WCDMA system with a queue.

We call this proposed buffer management technique the explicit rate change notification (ERCN) mechanism in this report. The ERCN mechanism can explicitly notify the TCP sender about the situation of the usage for the wireless link capacity by changing the buffer capacity limit. Hence, the proposed method can support the delay-sensitive TCP traffic [10], such as interactive data traffic in the WCDMA system [11].

The rest of this report is organized as follows. Section II describes the background on the TCP behavior. Section III formulates the buffer management for the WCDMA system. Section IV introduces the proposed ERCN mechanism. Numerical results are shown in the section V. We give our concluding remarks in the section VI.

## II. BACKGROUND ON TCP BEHAVIOR

In this report, we focus our work on the TCP with ERCN mechanism. We discuss the TCP behavior in the WCDMA system as shown in the Fig. 1, where, a TCP connection is established between TCP sender and mobile terminal with an intermediate base station. The wire link between TCP sender and the base station can have high capacity, while the wireless link between the base station and the mobile terminal has a very limited relatively. The base station has a queue for buffering the TCP segments and controls the WCDMA physical layer data rate. The variant WCDMA physical data rates would affect the performance of the TCP flow between the base station and the mobile terminal. In this work, we only consider the downlink transmission. Fig. 2 describes the variations of congestion window (CWND) size when sending TCP traffic. In Fig. 2, a round is defined as the duration of sending the first packet of  $W$  packets until the ACK is received, where  $W$  is the current CWND size. The reception of the ACK represents the end of the current round and the beginning of the next round. The CWND size is changed in each round according to the



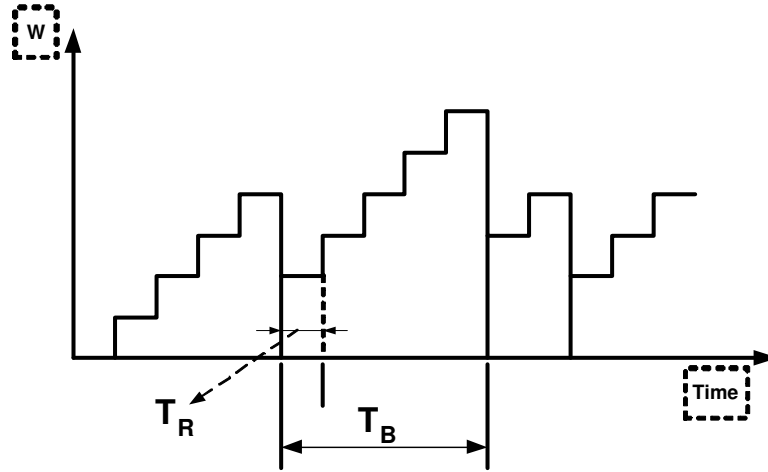


Fig. 2. The variation of CWND size in the TCP with ERCN.

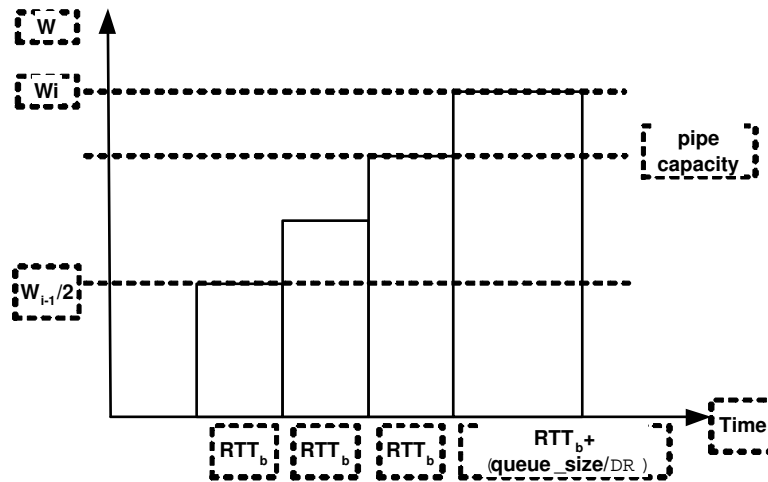


Fig. 3. The variations of the CWND size in a TCP block.

additive increase multiplicative decrease (AIMD) mechanism. In the AIMD mechanism, the CWND size is increased by one segment when receiving ACK, and is decreased by half when receiving the rate change notification. Next, we define the block period ( $T_B$ ) as the duration between two reductions of the CWND size, which is resulted from the notification of rate change sent from the base station.

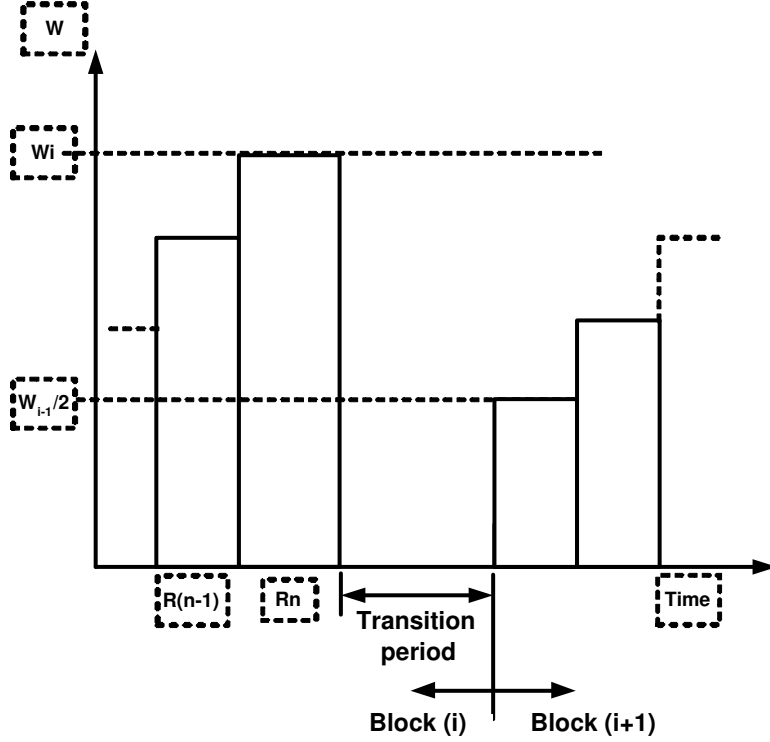


Fig. 4. The transition period between two blocks.

### III. MOTIVATION AND PROBLEM FORMULATION

#### A. The Relations of Queue Size, Throughput and Delay

Now we derive the TCP throughput in terms of queue size in the base station. Fig. 3 shows the variations of the CWND size for a TCP block, which has four rounds. Let  $W_i$  be the CWND size in the last round of the  $i$ -th block. According to the AIMD mechanism, the CWND size of the first round in the  $i$ -th block is  $W_{i-1}/2$ . Thus, there are  $(W_i - 0.5W_{i-1} + 1)$  rounds in the  $i$ -th block. Consequently, the total number of segments in the  $i$ -th block becomes  $(W_{i-1}/2 + W_i)(W_i - 0.5W_{i-1} + 1)/2$ . Denote  $B$  the throughput in a block.

$$B = \left( \frac{(W_{i-1}/2 + W_i)(W_i - 0.5W_{i-1} + 1)}{2} \right) \frac{1}{T_B}. \quad (1)$$

Denote  $\bar{W}$  the average number of segments in a block.

Then  $B$  can be approximate by

$$B = \frac{\frac{3}{8}\bar{W}^2 + \frac{3}{4}\bar{W}}{T_B}. \quad (2)$$

Now from Fig. 3,  $T_B$  will increase when the CWND size is larger than the link pipe capacity (PC). In such a situation, the TCP segments are buffered in the queue, and queue size increase the TCP delay from  $RTT_b$  to  $(RTT_b + queue\_size/R)$ , where the  $RTT_b$  is the round trip time when no segments be buffered in the base station and the R is the data rate of the radio link [3] and [12]. There exist a transition period in a block duration (see Fig. 4). During the transition period, the TCP sender receives a number of  $\bar{W}/2$  ACKs on average without sending any segment and the queue size will decrease  $\bar{W}/2$ . Referring to Fig. 3, we have

$$T_B = RTT_b \left( \frac{\bar{W}}{2} + 1 \right) + \frac{Q_{total}}{R}. \quad (3)$$

Denote  $Q_{total}$  the total queue size in a block. As mentioned above, we can express the  $\bar{Q}$  as

$$\bar{Q} = Q_{total} \frac{1}{\frac{\bar{W}}{2} + 1}, \quad (4)$$

Denote  $\bar{Q}$  the average queue size in a round.

Substituting (3) and (4) into (2), we can obtain

$$B = \frac{\frac{3\bar{W}}{4}}{RTT_b + \left( \bar{Q} + \frac{PC}{\bar{W}+2} \right) \frac{1}{R}}. \quad (5)$$

We also drive the average round trip time as

$$\overline{RTT} = RTT_b + \frac{\bar{Q}}{R}. \quad (6)$$

We assume the  $RTT_b$  and R are known by the base station. From (5) and (6), if we get the relation between  $\bar{W}$  and  $\bar{Q}$ , we can derive the B and  $\overline{RTT}$  as a function of  $\bar{Q}$ . Now, we consider two situations.

In the case of  $(PC < \bar{W} < 2PC)$ , we have

$$\bar{Q} = \frac{(1 + \bar{W} - PC)(\bar{W} - PC)}{2} \frac{1}{\frac{\bar{W}}{2} + 1}, \quad (7)$$

and

$$\bar{W} = \frac{-A + \sqrt{A^2 - 4(PC^2 - PC - 2\bar{Q})}}{2}, \quad (8)$$

where denote A is

$$A = -2PC - \bar{Q} + 1. \quad (9)$$

In the second case ( $2PC \leq \bar{W}$ ), we have

$$\bar{Q} = \frac{((\frac{\bar{W}}{2} - PC) + (\bar{W} - PC))(\frac{\bar{W}}{2} + 1)}{2} \frac{1}{\frac{\bar{W}}{2} + 1}, \quad (10)$$

and

$$\bar{W} = \frac{4(PC + \bar{Q})}{3}. \quad (11)$$

Fig. 5 shows the relations between the TCP throughput and the average queue size according to (5), (8) and (11). In the figure, the size of TCP segment is 500 bytes include the TCP header of 40 bytes; R is 256 kbps; the delay of the Internet is 100 ms; the delay of the radio link is (TCP segment size)/R equal to 15.6 ms; the  $RTT_b$  is 115.6 ms; the PC is  $((RTT_b) \times (R))$  equal to 29.6 kbits. Observing Fig. 5, when  $\bar{Q}$  increases from 5 kbits to 15 kbits, the TCP throughput only increase 10 kbits (2%).

Fig. 6 shows the relation between the TCP delay and average queue size based on (6), (8) and (11). When  $\bar{Q}$  increases from 5 kbits to 15 kbits, the TCP delay increase 40 ms (30%). From the above observation, we can control the queue size to reduce the TCP delay without decreasing too much throughput. In next section, we will discuss how to control the queue size.

### B. Control of the Queue Size

In our work, we apply the ECN mechanism [13] to control the queue size and Fig. 7 shows the relation between the  $\bar{Q}$  and  $p$ . The (12) is the detail behavior of the ECN.

$$p = \begin{cases} 0 & , \bar{Q} < Q_{min} \\ P_{max} \frac{\bar{Q} - Q_{min}}{Q_{max} - Q_{min}} & , Q_{min} \leq \bar{Q} < Q_{max} \\ 1 & , Q_{max} \leq \bar{Q} \end{cases} \quad (12)$$

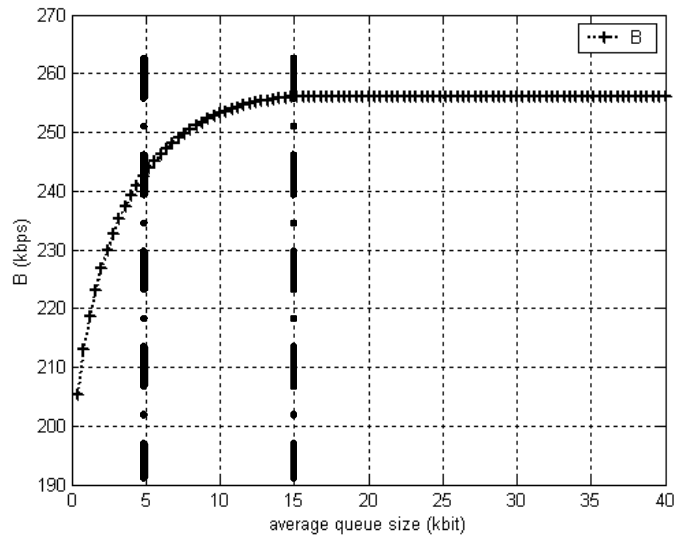


Fig. 5. The TCP throughput is a function of average queue size.

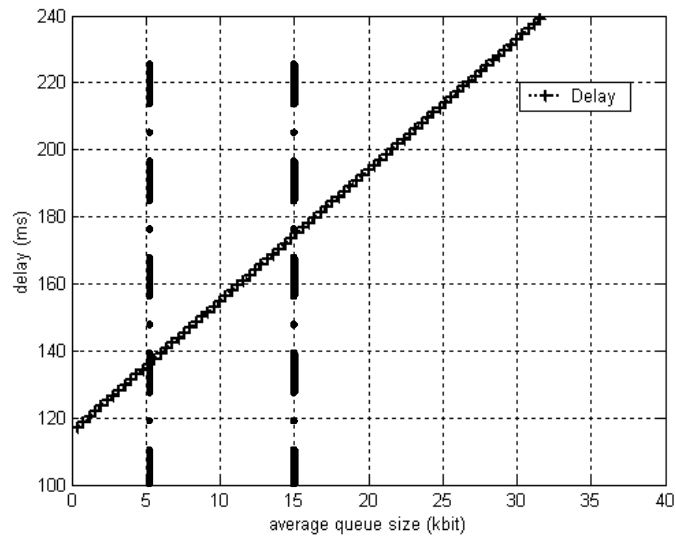


Fig. 6. The TCP delay is a function of average queue size.

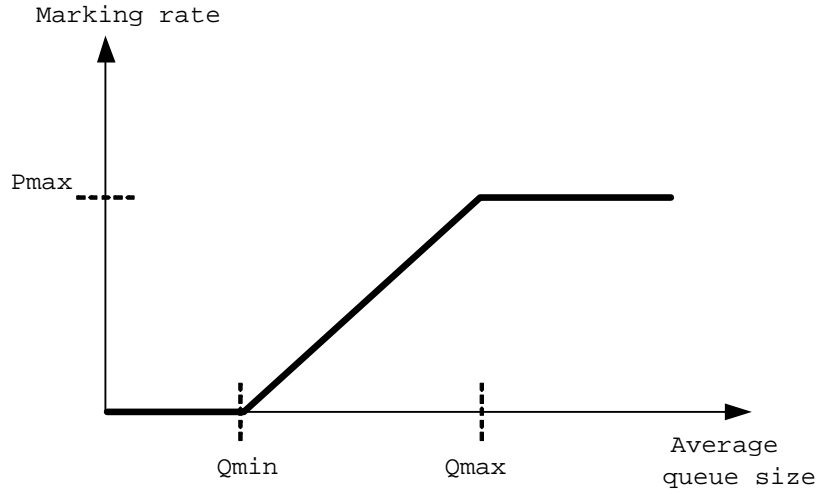


Fig. 7. The relation between average queue size and marking rate of the ECN.

Since the  $\bar{Q}$  size is relative to the marking probability of the ECN, we investigate the relation among  $\bar{Q}$ , marking probability and the throughput. As in [14],  $B$  can be expressed in terms of marking probability, i.e.,

$$B = \frac{\frac{1}{p} + \sqrt{\frac{8}{3p}}}{RTT(\sqrt{\frac{2}{3p}} + 1)}. \quad (13)$$

In each TCP round, when the segments in the TCP link are larger than  $PC$ , some segments will be buffered in the queue. Thus, the  $\bar{Q}$  is equal to

$$\begin{aligned} \bar{Q} &= B \times (\overline{RTT}) - PC, \\ \bar{Q} &= \frac{\frac{1}{p} + \sqrt{\frac{8}{3p}}}{\sqrt{\frac{2}{3p}} + 1} - PC. \end{aligned} \quad (14)$$

Fig. 8 shows the relation between average queue size and marking probability. As Fig. 5, we consider the case that  $R$  is 256 kbps and 512 kbps. We find that different data rate allocations have different relations between  $p$  and  $\bar{Q}$ . Now we try to modify the ECN to get different average queue size for the various data rate allocation systems. Specifically, we can change  $Q_{max}$  in the ECN mechanism to modify the marking probability.

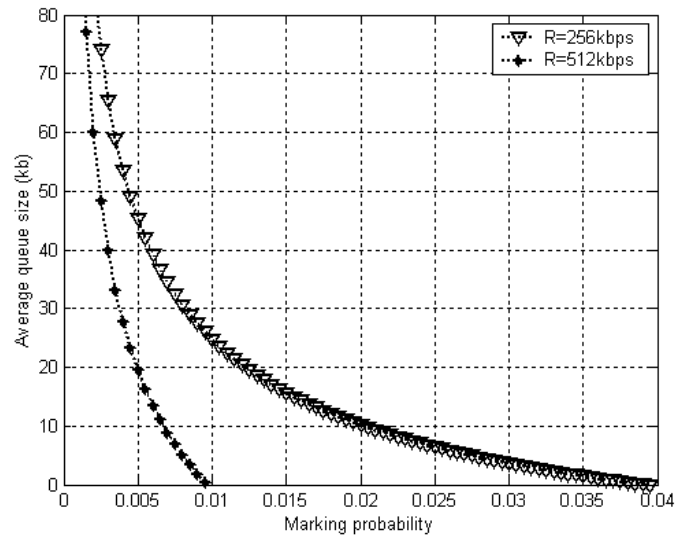


Fig. 8. The relation between queue size and marking probability for various data rates.

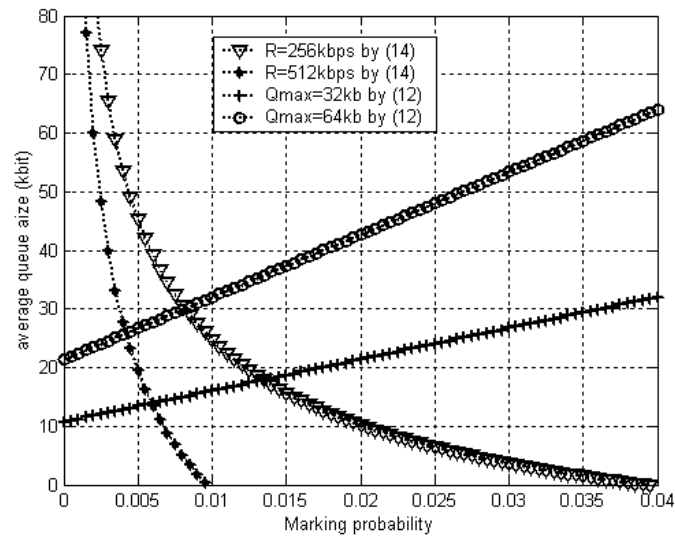


Fig. 9. Different ECN parameters settings can have different average queue size for different data rates.

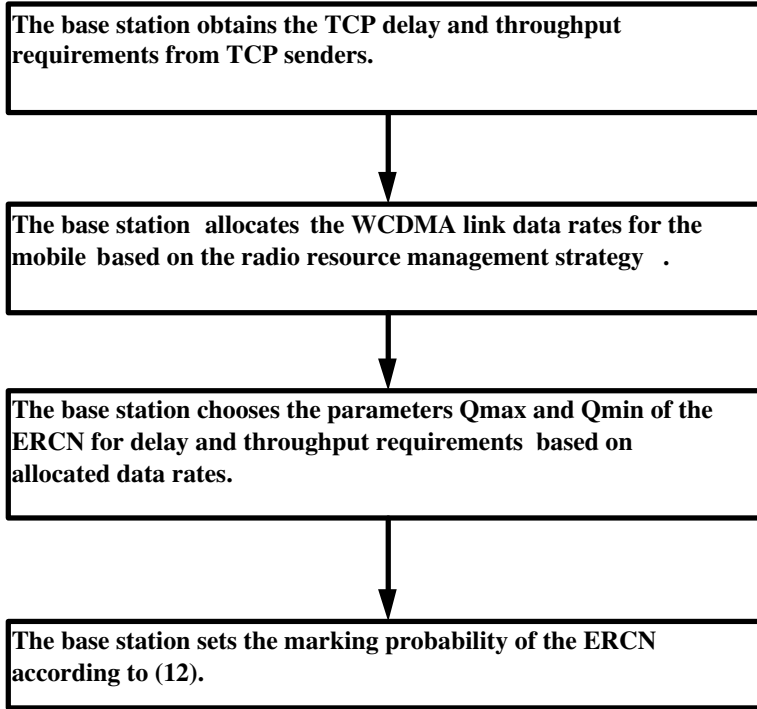


Fig. 10. Flow chart of the ERCN procedures.

In Fig. 9, we change the  $Q_{max}$  parameter of the ECN to show that different  $Q_{max}$  settings can have different average queue size for different data rates. For example is designed for  $R=512$  kbps, average queue size 25 kbits should be the  $Q_{max}=64$  kbits. For  $\bar{Q}=18$  kbits and  $R=256$  kbps, one should set  $Q_{max}=32$  kbits. In the next section, we will discuss how to modify the ECN to get different average queue sizes in the WCDMA system.

#### IV. THE PROPOSED ERCN MECHANISM

We propose Explicit Rate Change Notification (ERCN) by modifying the ECN mechanism. With ERCN, we can choose the queue size to control TCP delay and throughput for different WCDMA wireless link data rates. The detail ERCN procedures of the base station are shown in Fig. 10.

In the ERCN mechanism, we adapt the parameters  $Q_{max}$  and  $Q_{min}$  for different data rates as Fig. 11.

First, we set the  $Q_{min}$  as  $\frac{Q_{max}}{3}$  as recommended by [6]. Now, we analyze the relation between  $Q_{max}$  and  $\bar{Q}$ . From (14), we obtain the relation between  $p$  and  $\bar{Q}$ .



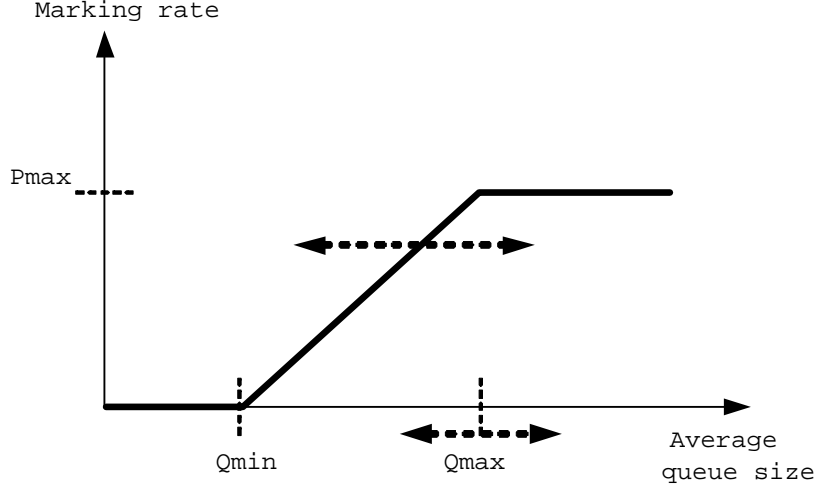


Fig. 11. The ERCN adapts the  $Q_{max}$  dynamically to different WCDMA air link data rates.

$$p = \left( \frac{-\sqrt{\frac{2}{3}}(\bar{Q} + PC - 2) + Y}{2(\bar{Q} + PC)} \right)^2. \quad (15)$$

We denote  $Y$  as

$$Y = \sqrt{\frac{2}{3}(\bar{Q} + PC - 2)^2 + 4(\bar{Q} + PC)}. \quad (16)$$

From (12) and (15), we obtain

$$Q_{max} = \frac{3(P_{max})(\bar{Q})}{2\left(\left(\frac{-\sqrt{\frac{2}{3}}(\bar{Q} + PC - 2) + Y}{2(\bar{Q} + PC)}\right)^2\right) + P_{max}}. \quad (17)$$

Notice that the relations between  $\bar{Q}$ ,  $\overline{RTT}$  and  $B$  are shown in (5), (6), (8) and (11). Hence, we can obtain the relations between  $Q_{max}$ ,  $\overline{RTT}$  and  $B$ .

## V. NUMERICAL RESULTS

### A. The Simulation Model

We applied the enhanced UMTS radio access network extensions (EURANE) [15] for ns-2 (network simulator version 2.26) [16] to simulate the system shown in Fig. 12. First, we describe the nodes in Fig. 12.

The mobile is a mobile device in the WCDMA system. There are a radio interface between the mobile and radio network. The radio resource control (RRC) functions of the WCDMA network can dynamically adjust the transmission bandwidth for the variant radio channel condition.

The base station of the WCDMA system connects the mobile and the WCDMA network. The base station communicates with radio network controller (RNC) in wide bandwidth wire link and communicates with the mobile in narrow air link bandwidth which is the bottleneck of the TCP link. Therefore, the TCP segments will be buffered in the queue of the base station. In the wire interface between base station and RNC, the transmission blocks are the radio link control (RLC) blocks.

The RNC controls the WCDMA radio link traffics over the UMTS network. It arranges the RLC channels for each mobile and connected to the the gateways which are connected to the Internet.

The TCP source is a node attached in the Internet. The TCP agent version is the TCP New Reno.

The delay and bandwidth information of the simulation model are described in Table. I. There, we only show the fixed wire link information. About the wireless link information, as mentioned, the wireless interface bandwidth between the base station and the mobile will be dynamically allocated by the RRC function and the allocated bandwidth also impacts the propagation delay.

TABLE I  
THE CONNECTION INFORMATION OF THE SIMULATION MODEL.

	to node	bandwidth (Mbps)	delay (ms)
TCP source	gateways	100	30
gateways	RNC	622	10
RNC	basestaion	622	15

### B. The Accuracy Validation of the Analysis

We establish a TCP flow between the TCP source and the mobile, in which base station is equipped with the ERCN mechanism with the parameters in (12) :  $P_{max}=0.04$ ;  $Q_{max}=3Q_{min}$ . We consider the TCP segment of 500 bytes with 40 bytes of TCP header. Let the the radio link

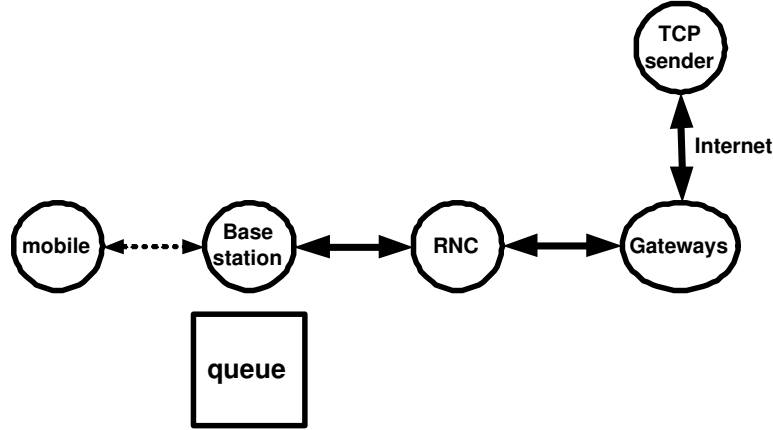


Fig. 12. UMTS reference architecture.

bandwidth is set to be 128 kbps and 256 kbps and the link layer channel is dedicated channel (DCH) with 20 ms transmission time interval. The RLC block size has 40 bytes payload and 2 bytes header. Because the 2 bytes RLC header impacts the analytical results we modify the (5) and (6) as follow.

$$B = D \times \frac{\frac{3\bar{W}}{4}}{RTT_b + (\bar{Q} + \frac{PC}{\bar{W}+2})\frac{1}{R}}, \quad (18)$$

where denote D is

$$D = \frac{RLC \text{ payload size}}{RLC \text{ payload size} + RLC \text{ header size}}. \quad (19)$$

$$\overline{RTT} = E \times \frac{TCP \text{ segment size}}{R} + RTT_b + \frac{\bar{Q}}{R}, \quad (20)$$

where denote E is

$$E = \frac{RLC \text{ header size}}{RLC \text{ payload size} + RLC \text{ header size}}. \quad (21)$$

Fig. 13 shows the relation of  $Q_{max}$  and the total delay for  $R=128$  kbps and 256 kbps. The analytical results are obtained according to (8), (11), (17) and (20) which match the simulation results well. Now consider the delay requirement is 260 ms set for example. From the figure, one can obtain  $Q_{max}=10$  kbits and 55 kbits for  $R=128$  kbps and 256 kbps, respectively. The achievable throughput by this option can be obtained by the next figure.

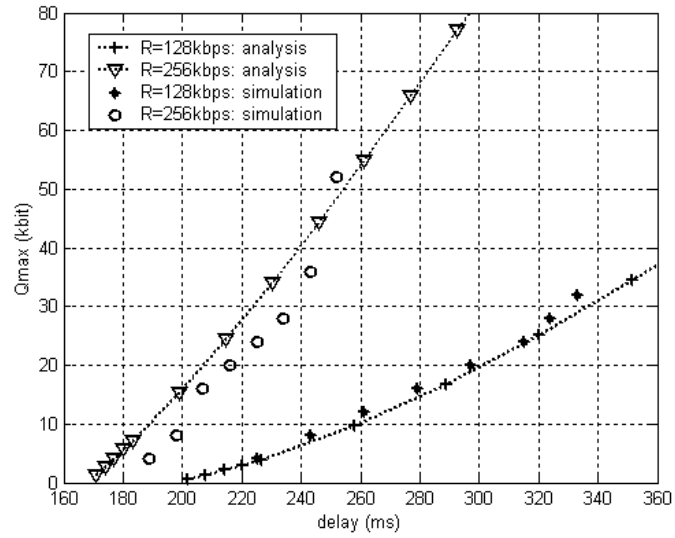


Fig. 13. The relation of Qmax and delay.

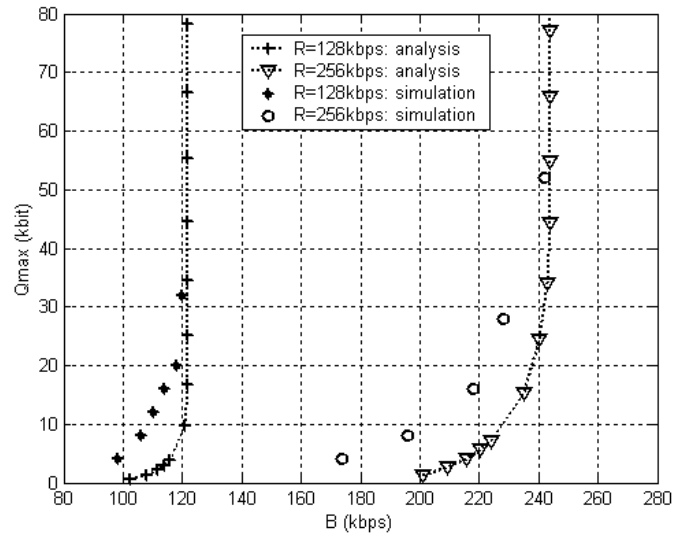


Fig. 14. The relation of Qmax and throughput.

Fig. 14 shows the relation of  $Q_{max}$  and throughput by simulations and analysis according to (8), (11), (17) and (18). One can find the simulation results match the analytical results well. From the figure, one can determine the expected throughput for different values of  $Q_{max}$ . For example, for  $Q_{max}=10$  kbits and 55 kbits, the throughput is 110 kbps and 240 kbps, respectively.

### C. The Comparison of ECN and ERCN

There, we have an example to compare the numerical results of the ECN and ERCN. In the WCDMA system, the 128 kbps service class is considered to support video telephone and various other data rate application [11]. We assume the RRC function allocated 128 kbps for the wireless interface between the base station and mobile in the wireless link setup state. After the wireless link setup state, for maintaining specific bit error rate, the WCDMA network can adapt the link data rate to variant wireless link condition, for example, increases the spread factor for worse link condition and the data rate will be 64 kbps, decreases spreading factor for better link condition and the data rate is 256 kbps. In this example, the ERCN dynamically adapts the  $Q_{max}$  to different wireless link data rates but the previous work ECN has a fixed  $Q_{max}$  in different wireless link data rates. Table. II shows the numerical results of the ERCN and ECN.

TABLE II  
THE PERFORMANCES OF THE ECN AND ERCN.

	Throughput ( <i>kbps</i> )	delay ( <i>ms</i> )	delay jitter $\times 10^4$ ( <i>ms</i> <sup>2</sup> )
ECN	115	307	1.73
ERCN	119	286	0.6

In Table. II, the throughput of the ERCN is better than the ECN, it is because when the wireless link data rate is 256 kbps the ERCN maintains a larger  $Q_{max}$  than the ECN, observing Fig. 14, the larger  $Q_{max}$  obtains the larger TCP throughput. The ERCN also has smaller average delay than the ECN, it is because the ERCN maintains a smaller  $Q_{max}$  than the ECN for 64 kbps wireless link data rate, observing Fig. 13, the smaller  $Q_{max}$  obtains smaller TCP delay. Since the ERCN maintains larger  $Q_{max}$  for larger wireless link data rate, smaller  $Q_{max}$  for smaller wireless link data

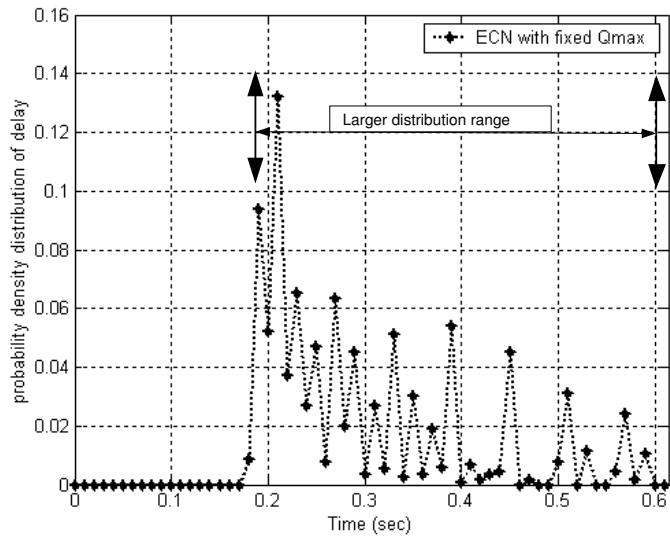


Fig. 15. The delay distribution of the ECN.

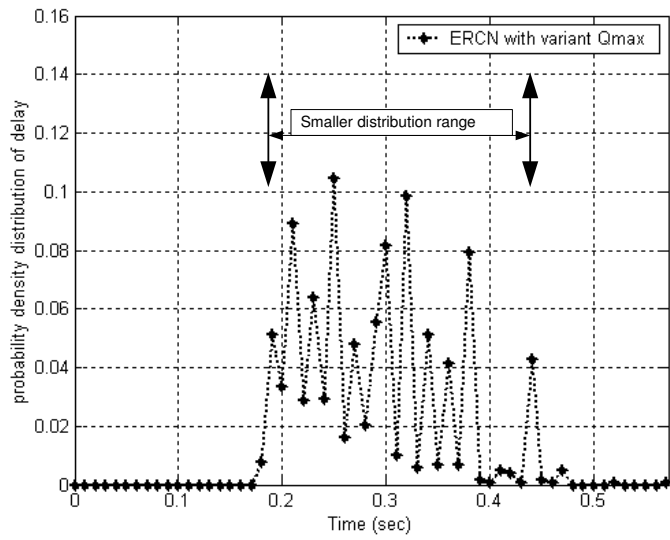


Fig. 16. The delay distribution of the ERCN.

rate, observing Fig. 15 and Fig. 16, the TCP delay distribution range of the ERCN is smaller than the ECN. Therefore, as the results of Table. II, the ERCN has better TCP delay jitter than the ECN.

## VI. CONCLUSIONS

In this report, we have analyzed the relation among the physical layer bandwidth, the TCP layer throughput, delay and queue size when delivering the TCP traffic in the WCDMA system. The developed analytical formulas can facilitate the design of queue sizes in the buffer of base station for different TCP throughput and delay requirements. Specifically, we propose an explicit rate change notification (ERCN) mechanism to dynamically change the queue size of the buffer in the base station based on the TCP/Physical cross-layer performance issues. Hence, when the radio link capacity in the physical layer is changed, the suitable queue size in the base station can be effectively adapted and the TCP sender can accordingly change the transmit data rates, thereby both the TCP delay and throughput requirements can be met in the varying radio channel conditions. In addition, we validate the accuracy of the analysis by simulation in the EURANE, the simulation results are similar to the results of analysis. Finally, we have numerical results to show the ERCN can improve the TCP throughput, delay and delay jitter when the WCDMA system adapts the wireless link data rate to the radio channel condition.

## REFERENCES

- [1] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," *IEEE Communications Magazine*, vol. 41, NO. 10, pp. 74 – 80, Oct 2003.
- [2] F. Lefevre and G. Vivier, "Understanding tcp's behavior over wireless links," *IEEE Vehicular Technology Conference*.
- [3] M. Sagfors, R. Ludwig, M. Meyer, and J. Peisa, "Buffer management for rate-varying 3g wireless links supporting tcp traffic," *IEEE Vehicular Technology Conference*, vol. 1, pp. 675 – 679, April 2004.
- [4] —, "Queue management for tcp traffic over 3g links," *IEEE Wireless Communications and Networking Conference*, vol. 3, pp. 1663–1668, March 2003.
- [5] H. Ekstrom and A. Schieder, "Buffer management for the interactive bearer in geran," *IEEE Vehicular Technology Conference*, vol. 4, pp. 2505–2509, April 2003.
- [6] S. Floyd, "Red wed page," Aug 2001. [Online]. Available: <http://www.aciri.org/floyd/red.html>
- [7] R. Stevens, *TCP/IP Illustrated*. Addison-Wesley, 1994, vol. 1.
- [8] M. Yavuz and F. Khafizov, "Tcp over wireless links with variable bandwidth," *IEEE Vehicular Technology Conference*, vol. 3, pp. 1322 – 1327, Sept 2002.
- [9] P. J. Ameigeiras, J. Wigard, and P. Mogensen, "Impact of tcp flow control on the radio resource management of wcdma networks," *IEEE Vehicular Technology Conference*, vol. 2, pp. 977 – 981, May 2002.

- [10] R. Cuny and A. Lakaniemi, "Voip in 3g networks: an end-to-end quality of service analysis," *IEEE Vehicular Technology Conference*, vol. 2, pp. 930–934, April 2003.
- [11] H. Holma and A. Toskala, *WCDMA FOR UMTS*. England: John Wiley and Sons, 2000.
- [12] M. Kwon and S. Fahmy, "Tcp increase/decrease behavior with explicit congestion notification (ecn)," *IEEE International Conference on Communications*, vol. 4, pp. 2335 – 2340, May 2002.
- [13] K. K. Ramakrishnan, S. Floyd, and D. Black, *The Addition of Explicit Congestion Notification (ECN) to IP*, RFC 3168, Sept 2001.
- [14] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, "Modeling tcp reno performance: a simple model and its empirical validation," *IEEE/ACM Transactions on Networking*, vol. 2, NO. 3, pp. 133 – 145, April 2000.
- [15] "Enhanced umts radio acces network extension for ns-2," Aug 2004. [Online]. Available: <http://www.ti-wmc.nl/eurane/>
- [16] "The network simulator ns2," Aug 2004. [Online]. Available: [www.isi.edu/nsnam/ns/](http://www.isi.edu/nsnam/ns/)



# Comparisons of Link Adaptation Based Scheduling Algorithms for the WCDMA System with High Speed Downlink Packet Access

Li-Chun Wang and Ming-Chi Chen

## I. INTRODUCTION

In order to satisfy the fast growing demand of the wireless packet data services, the concept of high speed downlink packet access (HSDPA) is proposed as an evolution for the wideband code division multiple access (WCDMA) system [1]. The goal of the WCDMA system with HSDPA is to support peak data rates from 120 kbps to 10 Mbps by adopting many advanced techniques, such as fast link adaptation, fast physical layer retransmission, and efficient scheduling techniques [2]. A fast link adaptation mechanism can enhance throughput performance by adapting modulation and coding schemes in the rapidly changing radio channel. The hybrid automatic repeat request (HARQ) technique can improve the radio link performance by combining retransmitted packets with previous erroneous packets. Scheduling is the key to achieving fairness in a shared channel for multiple users. Basically, a scheduling algorithm is to select a most suitable user to access the channel in order to optimize throughput, fairness, and delay performances.

Recently, scheduling has attracted much attention for wireless data networks because it can exploit the multi-user diversity [3] [4]. In the traditional voice-oriented cellular network, the fluctuation of the fast fading is viewed as a drawback. However, channel variations can be also beneficial to the wireless *data* network. Because data services can tolerate some delay, a scheduling mechanism can be designed to select the user with the highest channel peak and serve one user at a time in a time-multiplexing fashion. With a higher channel peak, a more efficient modulation/coding scheme can be applied to enhance data rates. In general, a larger dynamic range of channel variations can yield higher channel peaks, thereby delivering larger multi-user diversity gain.

In addition to throughput optimization, service delay and fairness are another two important aspects needed to be taken into account in designing a good scheduling algorithm. The reason why wireless scheduling can improve system throughput lies in the fact that data services can tolerate certain level of delay. Nevertheless, a constraint on the maximum service delay is still necessary. In wireless systems, mobile users are located at different locations with different channel conditions. If a scheduler always selects the user with the best channel condition, some users at the cell boundary, for instance, may never have a chance to access the system. Hence, how to design a scheduling algorithm to achieve high throughput subject to delay and fairness constraint becomes a crucial and challenging issue for the wireless data network.

In the literature, according to the considered channel models, wireless scheduling algorithms can be categorized into two major types. First, the wireless scheduling algorithms in [5] [6] [7] considered a two-state on-off Markov channel model. Because of simplicity, the two-state Markov channel model is suitable to examine the fairness performance of scheduling algorithms. However, using the simple two-state Markov channel has limitations in capturing actual radio channel characteristics. Second, some wireless scheduling algorithms like [1] [8] [9] [11] considered a more practical radio channel model with the emphasis on exploiting the multi-user diversity. In [1], the maximum carrier to interference ratio (C/I) scheduler is designed to assign the channel to the user with the best C/I. Obviously, the maximum C/I scheduler fully utilizes the multi-user diversity, but is an unfair scheduling policy. In contrast to the maximum C/I scheduler, a fair time scheduler (or called the round robin scheduling algorithm [8]) allocates the channel to users in sequence with equal service time. Clearly, the fair time scheduler does not consider the channel effect. In [2] [8] [9] [10], the proportional fair scheduler was proposed for the IS-856 system and the WCDMA system. The proportional fair scheduler improves the fairness performance of the maximum C/I scheduler at the cost of lowering system throughput. However, in [11] it was pointed out that the proportional fair scheduling algorithm does not account for service delays. Thus, the authors in [11] proposed the exponential rule scheduler to improve the delay performance of the proportional fair scheduler. It was proved that the exponential rule scheduler is throughput optimal in the sense of making a service queue stable [12]. In [13], it was concluded that the exponential rule scheduler is superior to the proportional fair scheduler since it provides excellent latency performance even with a slightly lower system throughput. Nevertheless, the factor of queue length is still not explicitly

considered in the exponential rule scheduling policy.

To our knowledge, a wireless scheduling policy with consideration of all the factors of channel variations, service delay, and queue length is still lacking in the literature. Consequently, we are motivated to develop such a wireless scheduling algorithm. The contributions of this work are two folds. The first contribution of this work is to propose a queue-based exponential rule scheduler to explicitly take account of all the factors consisting of queue length, service delay, and channel variations. We find that the queue-based exponential rule scheduler can further improve the fairness performance as compared to the original exponential rule scheduler, while maintaining the same throughput and delay performances.

Second, in the context of multi-type services, we suggest a fairness index to evaluate the fairness performance of the link adaptation based wireless scheduling algorithms, i.e., the maximum C/I, proportional fair, and exponential rule schedulers. This fairness index is modified from the one used in the two-state Markov channel based wireless scheduling algorithms [5] [6] [7] [14]. Interestingly, we find the fairness of current link adaptation based wireless scheduling algorithms [1] [8] [9] [11] is not clearly specified. Thus, we are motivated to adopt a formal fairness index to evaluate the fairness performance of these link adaptation based wireless scheduling algorithms. Using this fairness index to compare scheduling algorithms is important, especially supporting multi-type services. For example, the fair time (or round robin) scheduler is viewed as the performance upper bound in terms of fairness for most link adaptation based wireless scheduling algorithms. However, this upper bound is only valid under the assumption that all users subscribe the same type of service any time. In the case of multi-type services, the fair time scheduler can only guarantee the equal access time for multiple users, but not ensure to satisfy the different requirements for different users. Thus, even the fair time scheduler may not be the fairest scheduling algorithm in supporting the multi-type services. Thus, to support multi-type services, it is important to re-evaluate the fairness of these link adaptation based wireless scheduling algorithms based on a formal fairness index. Our simulation results show that in the time-multiplexing fashion, the fairness performance of the exponential rule scheduler is very close to that of the fair time scheduler and both schemes are superior to the proportional fair scheduler. On the other hand, in the code-multiplexing fashion, we find that the exponential rule scheduler is only slightly better than the proportional fair rule, and both schemes are worse than the fair time scheduler.

TABLE I  
MODULATION AND CODING SCHEMES IN THE HSDPA CONCEPT

Modulation and coding schemes (MCS)	Modulation	Effective code rate
MCS 1	QPSK	1/4
MCS 2	QPSK	1/2
MCS 3	QPSK	3/4
MCS 4	16-QAM	1/2
MCS 5	16-QAM	3/4

The rest of this part of report is organized as follows. Section II briefly introduces the background of the HSDPA concept in the WCDMA system. Section III describes existing link adaptation based scheduling algorithms. We discuss our new proposed queue-based exponential rule scheduling algorithm in Section IV. Simulation results are presented in Section V. Finally, we give our concluding remarks in Section VI.

## II. HIGH SPEED DOWNLINK PACKET ACCESS

In this section, we introduce three key technologies in implementing the HSDPA concept, including adaptive modulation and coding, fast scheduling, and multi-code assignment.

### *A. Adaptive Modulation and Coding*

Adaptive modulation and coding is a link adaptation technique to adapt transmission parameters to the time varying channel conditions. The basic principle in applying adaptive modulation and coding is to assign higher order modulation and higher code rate to the users in favorable channel conditions, whereas allocate lower order modulation and lower code rate to the users in unfavorable channel conditions. To obtain the channel conditions, the receiver is required to measure the channel conditions and feedback to the transmitter periodically, e.g., 2 msec of every transmission time interval (TTI) in HSDPA. The major benefit of adaptive modulation and coding is to deliver higher data rate to the user with better channel conditions, thereby increasing the average throughput of the cell. Table I lists the modulation and coding schemes used in HSDPA.

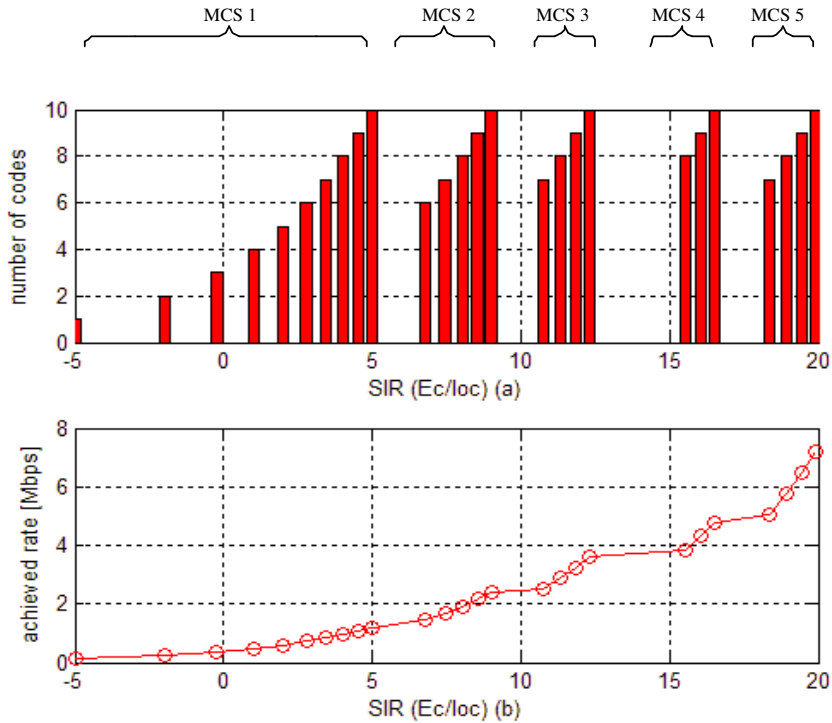


Fig. 1. The *hull* curve with five modulation and coding schemes and the multi-code operation of maximum 10 codes [1].

### B. Fair Scheduling

In the downlink shared channel, scheduling techniques can be used to exploit multi-user diversity by taking advantage of short-term channel variations of each user terminal. With scheduling the selected user is always served in the high channel peak or a constructive fade. When combined with adaptive modulation and coding, the scheduled user can therefore have a better chance to transmit at a higher rate with higher order modulation and higher code rate.

### C. Multi-Code Assignment

To achieve a higher data rate in HSDPA, adaptive modulation and coding technique needs to function together with multi-code assignment. Because the number of modulation and coding schemes is limited (e.g., only 5 formats in HSDPA), the dynamic range of selecting adaptive modulation and coding scheme may not be enough, e.g., 15 dB of dynamic range in an example of Fig. 51 in [1]. To increase the dynamic range of adapting transmission parameters to channel variations, selecting both the modulation/coding scheme and the number of codes is an effective

method. Figure 1 shows the *hull* curve of combining 5 modulation/coding schemes with the multi-code operation of 10 codes. As shown in the figure, the dynamic range of selecting transmission parameters for different  $E_c/I_{oc}$  (i.e., signal to interference ratio (SIR)) becomes -5 to 20 dB, which is 10 dB larger than the single code case of [1]. Generally speaking, it is better to increase the number of multi-code first before transiting to the next higher order of modulation/coding scheme based on a spectral efficiency viewpoint. Note that Fig. 1 in this report is produced by referring to Fig. 51 of [1], both of which do not consider HARQ. If ARQ is considered, we can straightforwardly use the hull curve of adaptive modulation and coding with ARQ in the single code operation (e.g., Fig. 50 of [1]) to produce a hull curve of the adaptive modulation/coding scheme with HARQ for the multi-code operation.

### III. CURRENT LINK ADAPTATION BASED SCHEDULING ALGORITHMS

As mentioned, the key factor in determining the performance of the WCDMA system with HSDPA is the link adaptation based scheduling algorithm [1] [8] [9] [11]. We will evaluate these scheduling algorithms in terms of three performance metrics: system throughput, delay, and fairness. In order to achieve the multi-user diversity, a good scheduling algorithm should take account of channel variations. The more the multi-user diversity, the higher the system throughput. However, for some delay-sensitive services, such as streaming video, a good scheduler also needs to pay attention to the latency performance. Furthermore, from an individual user standpoint, it is desirable to receive the service as fair as others. Thus, to design a good scheduling algorithm, we need to make a better tradeoff design between these three performance metrics.

Before we detail the scheduling algorithms, let us first define the following notations:

- $k$ : the index of the  $k^{th}$  transmission time interval (TTI).
- $\gamma_i(k)$ : the short-term SIR of user  $i$  averaged in the  $(k - 1)^{th}$  TTI.
- $\overline{\gamma_i(k)}$ : the long-term average SIR of user  $i$  observed in  $[(k - T), k]$ , where  $T$  is the length of sliding window in terms of the number of TTIs.
- $d_i(k)$ : the delay time for the packet waiting in the head of line (HOL) TTI before getting the service for user  $i$ .
- $q_i(k)$ : the queue length of user  $i$  at the beginning of the  $k^{th}$  TTI.

### A. Maximum C/I Scheduler

The maximum C/I scheduler always selects the user with the best carrier-to-interference ratio (C/I). At the beginning of each TTI, the scheduler compares the C/I levels of all active users and grants the channel access to the user with the highest C/I level. Specifically, the maximum C/I scheduler will select the user  $j$  in the  $k$ -th TTI if

$$j = \arg\{\max_i \gamma_i(k)\}. \quad (1)$$

Obviously, the maximum C/I scheduler is the upper bound in terms of system throughput because it can achieve the maximum multi-user diversity. However, those who are located far from the base station may feel unfair owing to the poor radio link condition. This unfairness phenomenon makes the maximum C/I scheduler in practical.

### B. Fair Time Scheduler

At any scheduling instant, the fair time scheduler serves all non-empty source queues in a round-robin fashion. That is, the fair time scheduler will schedule user  $j$  in the  $k$ -th TTI if

$$j = \text{mod}((k - 1), N) + 1, \quad (2)$$

where  $\text{mod}(\cdot)$  denotes the modulus operator and the  $N$  is the number of active users in the system. One can easily find that this fair time scheduling algorithm is independent of the channel characteristics and thus does not exploit the multi-user diversity at all. However, this scheduler has the best delay performance and is much fairer than the maximum C/I scheduler. Note that the fairness here is defined from the viewpoint of equal access probability. In this report, we will adopt another fairness index used in the two-state Markov channel based wireless scheduling algorithms [14] to examine the fairness performance of the scheduling algorithms.

### C. Proportional Fair Scheduler

The proportional fair algorithm was proposed to the HDR system [9]. The basic idea of this algorithm is to select a scheduled user based on the ratio of the short-term SIR over the long-term averaged SIR value with respect to each active user. The proportional fair scheduler takes advantage of the temporal variations of the channel to increase system throughput, while maintaining certain

fairness among all active users. Specifically, the proportional fair scheduler will schedule user  $j$  in the  $k$ -th TTI if

$$j = \arg\left\{\max_i \frac{\gamma_i(k)}{\overline{\gamma_i(k)}}\right\}. \quad (3)$$

Based on the above criterion, the  $\overline{\gamma_i(k)}$  is the average SIR measured over a sliding window as follows,

$$\overline{\gamma_i(k+1)} = \begin{cases} (1 - \frac{1}{T})\overline{\gamma_i(k)} + \frac{1}{T}\gamma_i(k) & \text{if user } i \text{ is scheduled,} \\ (1 - \frac{1}{T})\overline{\gamma_i(k)} & \text{if user } i \text{ is not scheduled.} \end{cases}$$

Note that the proportional fair scheduler does not consider the delay issue in each user's service queue and thus has poor delay performance.

#### D. Exponential Rule Scheduler

The exponential rule is a modified version of proportional fair. This scheduler further takes delay issue into consideration [11] [13]. This policy tries to balance the weighted delay of all active users when the differences of weighted queue delay among users become significant. For the  $k^{\text{th}}$  TTI, the exponential rule scheduler will choose user  $j$  if

$$j = \arg\left\{\max_i a_i \frac{\gamma_i(k)}{\overline{\gamma_i(k)}} \exp\left(\frac{a_i d_i(k) - \overline{ad(k)}}{1 + \sqrt{\overline{ad(k)}}}\right)\right\}, \quad (4)$$

where

$$\overline{ad(k)} = \frac{1}{N} \sum_{i=1}^N a_i d_i(k), \quad (5)$$

and  $a_i > 0$ ,  $i = 1, \dots, N$  are selected weights to characterize the desired quality of service.

To obtain the intuition behind the exponential rule scheduler, let us focus on the exponent term of (4). If an active user has a larger weighted delay than others by more than  $\sqrt{\overline{ad(t)}}$ , then the exponent term will become dominant and even exceed the impact of channel effect. In such a case, this user will get higher priority of the access opportunity. On the other hand, for small differences in the weighted delay, the exponent term will approach to 1 and the exponential rule scheduler is exactly the same as the original proportional fair rule scheduler. Note that the factor 1 in the denominator of (4) is to prevent the exponent term from increasing dramatically when  $\sqrt{\overline{ad(t)}}$  is



too small. In brief, the exponential rule scheduler incorporates both the effect of channel variations and service delay and aims to further improve the performance over the proportional fair scheduler.

#### IV. PROPOSED QUEUE-BASED EXPONENTIAL RULE SCHEDULER

In this section, we will first define the fairness and then propose an improved version of exponential rule scheduler.

##### A. Fairness Definition

Based on the traditional fluid fair queueing in wireline networks, backlogged flows are served in proportion to their weighted rate. Mathematically, for any time interval  $[t_1, t_2]$ , the channel capacity allocated to flow  $i$ , denoted as  $W_i(t_1, t_2)$ , should satisfy the following condition [14] [15]:

$$\left| \frac{W_i(t_1, t_2)}{r_i} - \frac{W_j(t_1, t_2)}{r_j} \right| = 0, \quad \forall i, j \in B(t_1, t_2), \quad (6)$$

where  $r_i$  and  $r_j$  are the weights of flows  $i$  and  $j$ , respectively, and  $B(t_1, t_2)$  is the set of backlogged flows during  $(t_1, t_2)$ .

However, this condition in terms of bits can not be maintained in a practical packet switched network. The goal of the packetized fair queueing algorithm is to minimize the difference of  $|W_i(t_1, t_2)/r_i - W_j(t_1, t_2)/r_j|$ . Therefore, we define a fairness index as follows,

$$FI = \frac{1}{l} \left| \frac{W_i(t_1, t_2)}{r_i} - \frac{W_j(t_1, t_2)}{r_j} \right|, \quad (7)$$

where  $l$  is a normalization factor of the packet size. In wireless networks, many wireless scheduling algorithms, such as IWFQ [5], CIF-Q [6], and WFS [7], were proposed to minimize the value of  $FI$  during a long period, i.e., achieving the long-term fairness. Here, we define the packet size as the number of data bits that can be transmitted at the minimum data rate during one transmission time interval (TTI).

##### B. Queue-Based Exponential Rule Scheduler

The exponential rule scheduler considers the effect of the delay time in the head of line (HOL) TTI. However, it does not *explicitly* incorporate the factor of queue length into the scheduling policy. By observing (7), we note that to achieve fairness not only the HOL delay is required to be

considered, but also the queue length. Consequently, we are motivated to propose a queue-based exponential rule scheduler as follows. In the  $k^{\text{th}}$  TTI, the proposed scheduler will choose user  $j$  if

$$j = \arg\left\{\max_i a_i \frac{\gamma_i(k)}{\gamma_i(k)} \exp\left(\frac{a_i d_i(k) - \overline{ad}(k)}{1 + \sqrt{\overline{ad}(k)}}\right) \exp\left(\frac{q_i(k) - \overline{q}(k)}{1 + \overline{q}(k)}\right)\right\}, \quad (8)$$

where

$$\overline{ad}(k) = \frac{1}{N} \sum_{i=1}^N a_i d_i(k), \quad (9)$$

and

$$\overline{q}(k) = \frac{1}{N} \sum_{i=1}^N q_i(k). \quad (10)$$

The basic idea of second exponent term in (8) is to balance the service queue length among multiple users. Moreover, in order to prevent the second exponent term from exceeding that of first exponent term, the denominator of the second exponent term does not take the square root as that of the first exponent term.

### C. Time-Multiplexing Fashion v.s. Code-Multiplexing Fashion

In HSDPA, there are five available modulation and coding schemes and sixteen orthogonal variable spreading factor (OVSF) codes [8]. There are two methods to implement the multi-code operation. One is to assign all available multi-codes to one user at a time in a pure time-multiplexing fashion. The other is to assign different active users to each code in a code-multiplexing fashion. For the code-multiplexing multi-code assignment, we can treat each code as a server and scheduling algorithms will select suitable users for multiple servers simultaneously. Note that the total power budget in each base station is equally shared by all codes for both methods. The performance comparison of these two methods will be discussed in Section V.

## V. SIMULATION RESULTS

Through simulation, we compare the performances of different scheduling algorithms. Three types of simulation configurations are considered. First, we focus on the performance with the single code operation in the time-multiplexing fashion. In this case, the maximum data rate for each scheduled mobile user is 720 kbps. In the second simulation configuration, we focus on the

performance of the multi-code operation in the time-multiplexing fashion. With maximum 10 codes, the maximum achieved data rate in this configuration becomes 7.2 Mbps. For the third simulation configuration, we investigate the performance of the multi-code operation in the code-multiplexing fashion.

#### A. Simulation Model

We consider a cell layout with a center cell surrounded by other six neighboring cells. We only focus on the performance of the center cell and treat other cells as the sources causing the downlink interference. The mobile users are uniformly located in the center cell. We assume that 60% of the total base station transmitted power is allocated in supporting the HSDPA services. In HSDPA, there are 16 orthogonal codes with processing gain of 12 dB. Nevertheless, many other dedicated, shared and common channels may also need to use some codes. Thus, for the multi-code operation in HSDPA, the suggested maximum number of codes can be assigned to a user is 10 [8]. Table II lists other parameters used in our simulations [1]. We apply the *hull* curve of the link level simulation results obtained from [1] into our system level simulation. For each randomly located mobile user, the system will first calculate the corresponding received signal to interference ratio (SIR). It is assumed that the measured SIR in each mobile can be correctly sent back to the base station in time. We consider a mobile at a speed of 3 km/hr in a flat Rayleigh fading channel, or equivalently the maximum Doppler frequency  $f_d = 5.5$  Hz.

#### B. Single-Code Operation in the Time-Multiplexing Fashion

Figure 2 compares the fairness performance according to the fairness definition of (7) for all the considered schedulers. Based on (7), the smaller the fairness index, the fairer the system. Obviously, the scheduler of the least fairness is the maximum C/I scheduler. The fair time scheduler, proportional fair scheduler and exponential rule scheduler almost have the same fairness performance. It is noteworthy that the proposed queue-based exponential rule scheduler is fairer than all the other schedulers. Specifically, the proposed new scheduling algorithm improves the fairness index values by 31% to 50% as compared to the original exponential rule scheduler. Figure 3 compares the throughput performance of the proposed scheduling algorithm with four existing scheduling algorithms. For the maximum C/I scheduler, one can find that when the number of

TABLE II  
SIMULATION PARAMETERS

Parameters	Explanation/Assumption
Cell layout	7 hexagonal cells
Cell radius	1.6 km
User location	Uniform distribution
Antenna pattern	Omni-direction
Propagation model	$L = 128.1 + 37.6 \text{ Log}_{10}(R)$
HS-DSCH power budget/the total Node-B power	60 %
Shadowing Std. deviation	8 dB
Correlation distance of shadowing fading	50 m
Carrier frequency	2 GHz
Base station total transmit power	44 dBm
Fast fading model	Jakes spectrum
Number of HS-DSCH multi-codes	10
Transmission time interval (TTI)	2 msec
Simulation duration	5000 TTIs

users in the system increases, the system throughput improves because more multi-user diversity gain are achieved. The fair time scheduler results in the lowest throughput, whereas the throughput of the proportional fair scheduler is between the maximum C/I scheduler and the fair time scheduler. Our proposed queue-based exponential rule scheduler performs the same as the original exponential rule scheduler, but has lower throughput than the proportional fair scheduler. Note that when the number of users per cell increases, the achieved average throughput of both the exponential rule scheduler and queue-based exponential rule scheduler slightly decrease because the effect of the exponent term becomes dominant for the case with a large number of users in the system. Although the proportional fair scheduler has better throughput, the exponential rule scheduler has better delay performance, which will be discussed next.

Figure 4 shows the delay performance of the considered schedulers in terms of the maximum

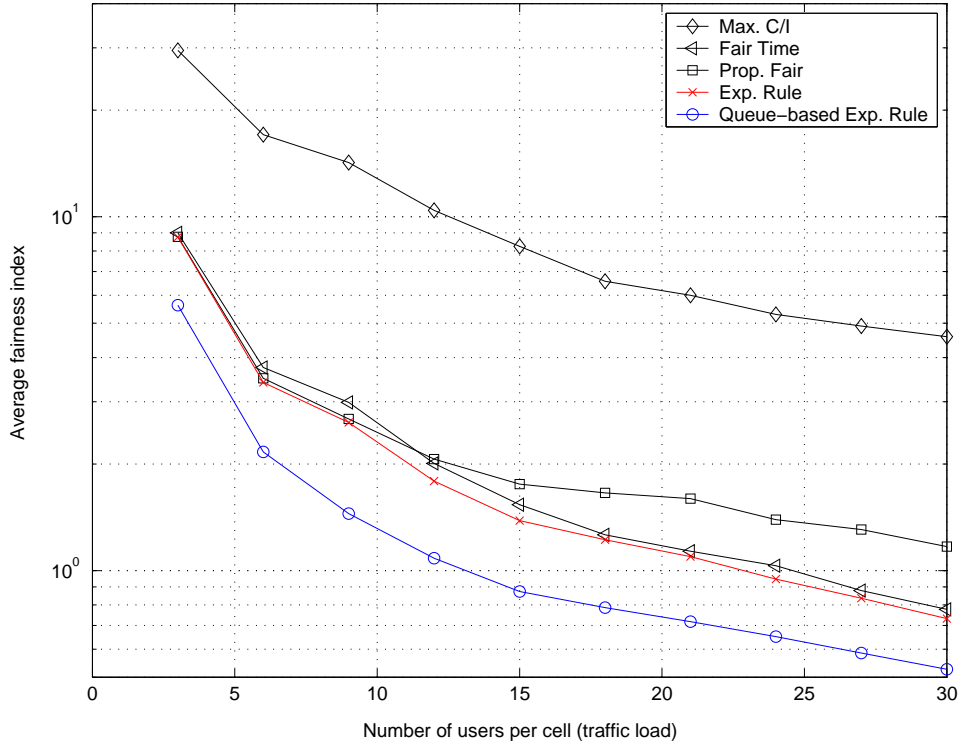


Fig. 2. The average of fairness index value for all schedulers without multi-code operation in a time-multiplexing fashion.

delay for the packet waiting in the HOL TTI. In general, higher traffic load causes longer delay because of more contenders. One can find that both the maximum C/I and proportional fair schedulers have very poor delay performance. As expected, the fair time scheduler performs the best. Note that both the proposed queue-based exponential rule scheduler and the original exponential rule scheduler perform well and have the same delay performance, while the proposed queue-based exponential rule scheduler has better fairness performance as shown in Fig. 2 previously.

### C. Multi-Code Operation in the Time-Multiplexing Fashion

Figures 5, 6 and 7 show the performances of fairness, system throughput and delay for different schedulers with multi-code operation in the time-multiplexing fashion, respectively. Basically, the trends of these performance curves are the same as in the single code case except that for the multi-code operation, the maximum achievable system throughput becomes 7.2 Mbps. Again, the fairness index of the new proposed algorithm improves ranged from 15% to 30% as compared to original exponential rule scheduler, while maintaining the same throughput and delay performance.

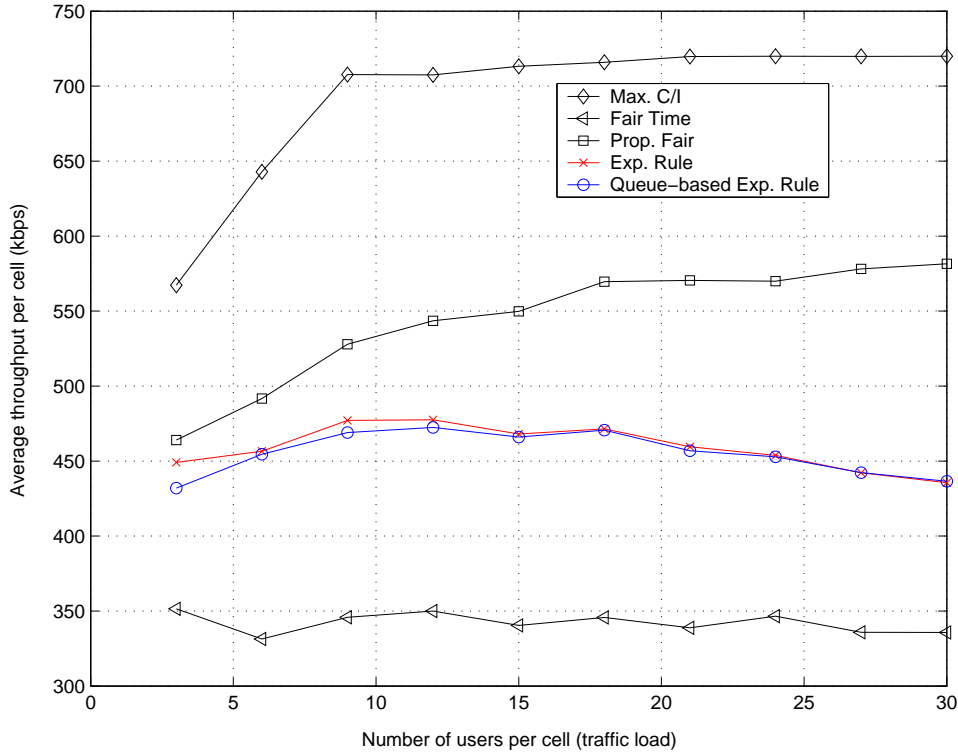


Fig. 3. System throughput performance of all schedulers without multi-code operation in a time-multiplexing fashion.

#### D. Multi-Code Operation in the Code-Multiplexing Fashion

Figure 8 shows the fairness performance of all schedulers in the case of code-multiplexing model. Interestingly, we find that the exponential rule scheduler is only slightly better than the proportional fair rule, and both schemes are worse than the fair time scheduler. Nevertheless, when we further evaluate the variance of the fairness index for different schedulers. As shown in Fig. 8, the variations of the fairness index of the queue-based exponential rule scheduler is much smaller than that of the fair time scheduler. This phenomenon implies that the proposed queue-based exponential can still have a high possibility to be fairer than the fair time scheduler even in the code-multiplexing case. In addition, the new proposed algorithm improved the fairness index value by 5% to 15% as compared to the original exponential rule scheduler. Figure 9 shows the system throughput performance of all schedulers with the multi-code operation in the code-multiplexing fashion. We consider more than 10 users in this simulation. One can find that the code-multiplexing

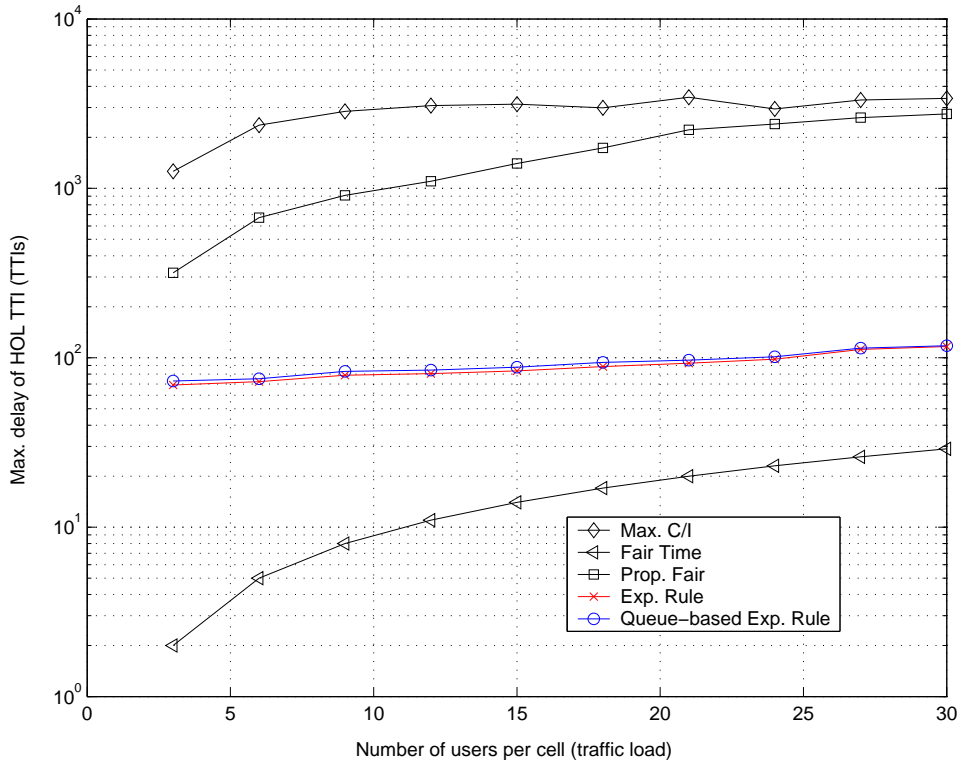


Fig. 4. The performance of maximum delay of all schedulers without multi-code operation in a time-multiplexing fashion.

fashion cannot fully take advantage of the multi-user diversity even for the maximum C/I method. This observation shows that assigning only one user at a time can achieve the maximum system throughput, which was also mentioned in [4]. Specifically, comparing Fig. 6 with Fig. 9 in the case of the system load equal to 15 users, the proposed scheduling algorithm can provide 2.3 Mbps of system throughput in the time-multiplexing scheme, while in the code-multiplexing scheme the throughput decreases to 1.8 Mbps. Figure 10 compares the corresponding delay performance of all schedulers in the code-multiplexing case. Because of more servers during each scheduling instant, the delay performance in the code-multiplexing is much better than that in the time-multiplexing fashion. For example, in the case of 30 users, the maximum delay of the queue-based exponential rule scheduler in the code-multiplexing fashion is reduced to 80 TTIs as compared to 100 TTIs in the time-multiplexing fashion. Thus, when comparing the performance between the second and third simulation configurations, we can conclude that the delay performance of the code-multiplexing scheme is better than the time-multiplexing at the expense of lower system throughput compared

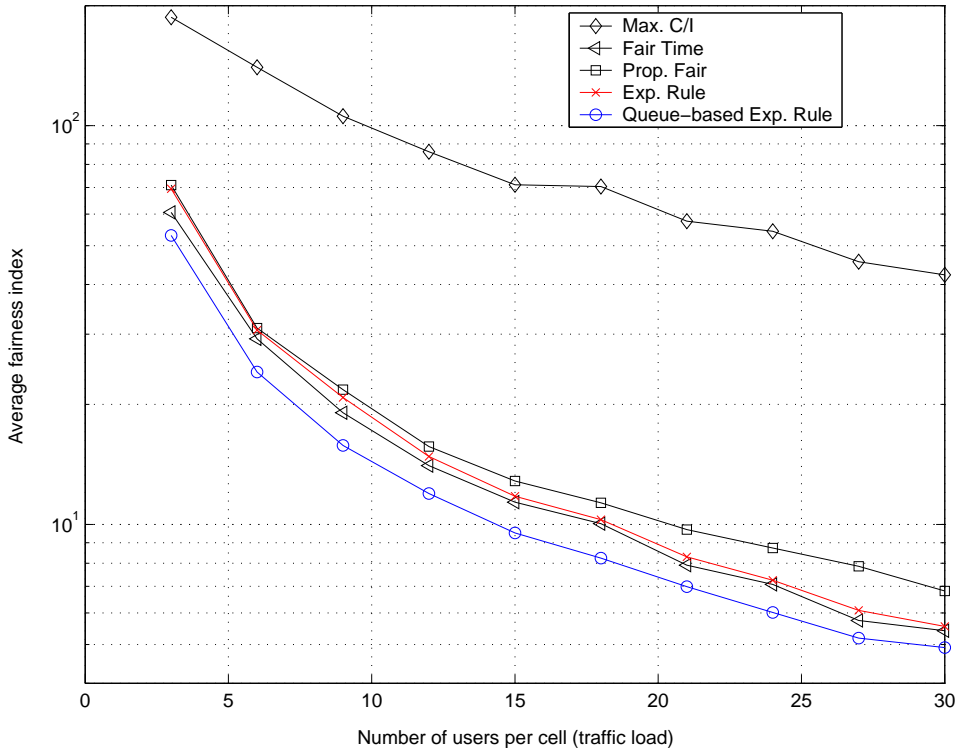


Fig. 5. The average of fairness index value for all schedulers with multi-code operation in a time-multiplexing fashion.

to the time-multiplexing scheme.

## VI. CONCLUSION

In this report, we adopt a fairness index to examine the fairness performance of current link adaptation based scheduling algorithms, including the maximum C/I, fair time, proportional fair, and exponential rule schedulers. Moreover, we have proposed a new queue-based exponential rule scheduler for the HSDPA system. As summarized in Table III, the proposed scheduling algorithm outperforms all the existing scheduling algorithms in terms of fairness for the time-multiplexing fashion. In the time-multiplexing case, the proposed queue-based exponential rule scheduler improves the fairness index in a range of 15% to 50% compared to the original exponential rule algorithm, while in the code-multiplexing case the improvement reduces to the range of 5% to 15%. For the code-multiplexing configuration, Table IV compares the schedulers considered with respect to fairness, delay, and throughput. Note that although the fairness performance of the proposed queue-based exponential rule scheduler is slightly worse than the fair time scheduler in



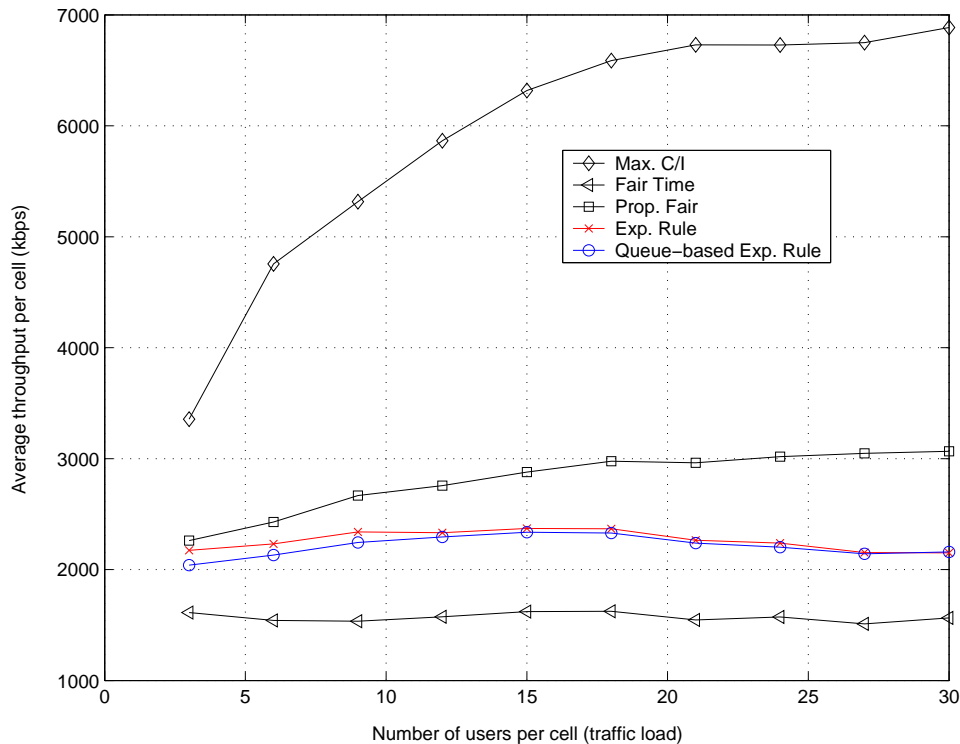


Fig. 6. System throughput performance of all schedulers with multi-code operation in a time-multiplexing fashion.

terms of the average fairness index value, the variations of the fairness index value of the proposed scheduler is smaller than the fair time scheduler, thereby still having possibility to be fairer than the fair time scheduler. It is noteworthy that in both the time-multiplexing and code-multiplexing cases, the proposed queue-based exponential rule scheduler and the original exponential scheduler improve the fairness and delay performances over the proportional fair scheduler at the cost of slightly degrading throughput. Of the comparison between code-multiplexing scheduling and time-multiplexing scheduling, we find that the time-multiplexing method is a better choice by taking all the factors of system throughput, service delay, and fairness into consideration.

## REFERENCES

- [1] 3GPP TR 25.848 V4.0.0, Physical Layer Aspects of UTRA High Speed Downlink Packet Access, Release 4, Mar. 2001.
- [2] T. E. Kolding, F. Frederiksen, and P.E. Mogensen, "Performance aspects of WCDMA systems with high speed downlink packet access (HSDPA)," in *Proceedings of 56th IEEE Vehicular Technology Conference (VTC 2002-fall)*, vol. 1, pp. 477 -481, Sep. 2002.

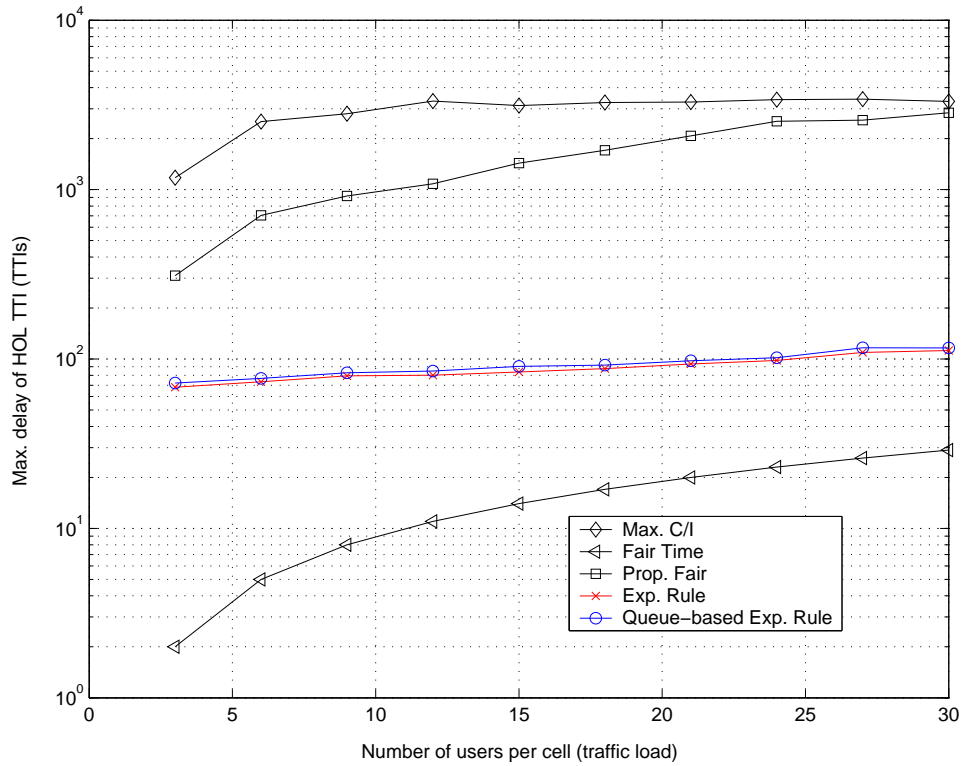


Fig. 7. The performance of maximum delay of all schedulers with multi-code operation in a time-multiplexing fashion.

TABLE III

COMPARISON OF LINK ADAPTATION BASED SCHEDULING ALGORITHMS IN TERMS OF FAIRNESS, THROUGHPUT, AND DELAY IN THE TIME-MULTIPLEXING FASHION

Scheduling policy	Fairness	Delay	Throughput
Maximum C/I	Poor	Poor	Excellent
Fair time	Good	Excellent	Poor
Proportional fair	Acceptable	Poor	Good
Exponential Rule	Good	Good	Acceptable
Queue-based exponential Rule	Excellent	Good	Acceptable

p.s. comparison level: excellent > good > acceptable > poor.

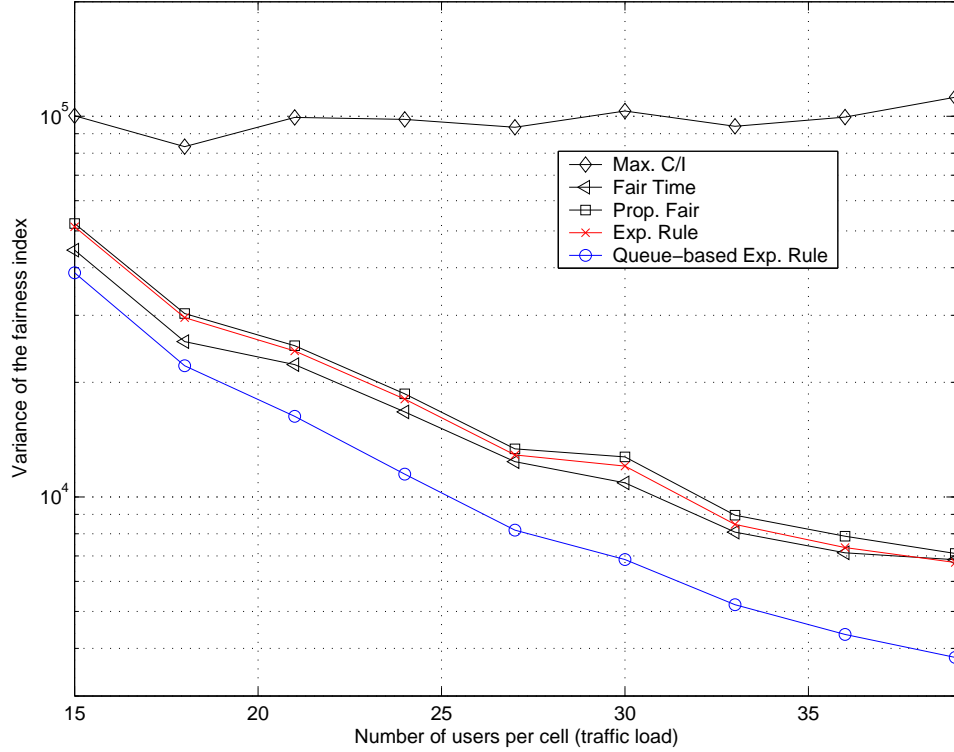


Fig. 8. The variance of fairness index value for all schedulers with multi-code operation in a code-multiplexing fashion.

TABLE IV

COMPARISON OF LINK ADAPTATION BASED SCHEDULING ALGORITHMS IN TERMS OF FAIRNESS, THROUGHPUT, AND DELAY IN THE CODE-MULTIPLEXING FASHION

Scheduling policy	Fairness	Delay	Throughput
Maximum C/I	Poor	Poor	Excellent
Fair time	Excellent	Excellent	Poor
Proportional fair	Acceptable	Poor	Acceptable
Exponential Rule	Acceptable	Good	Acceptable
Queue-based exponential Rule	Good	Good	Acceptable

p.s. comparison level: excellent > good > acceptable > poor.

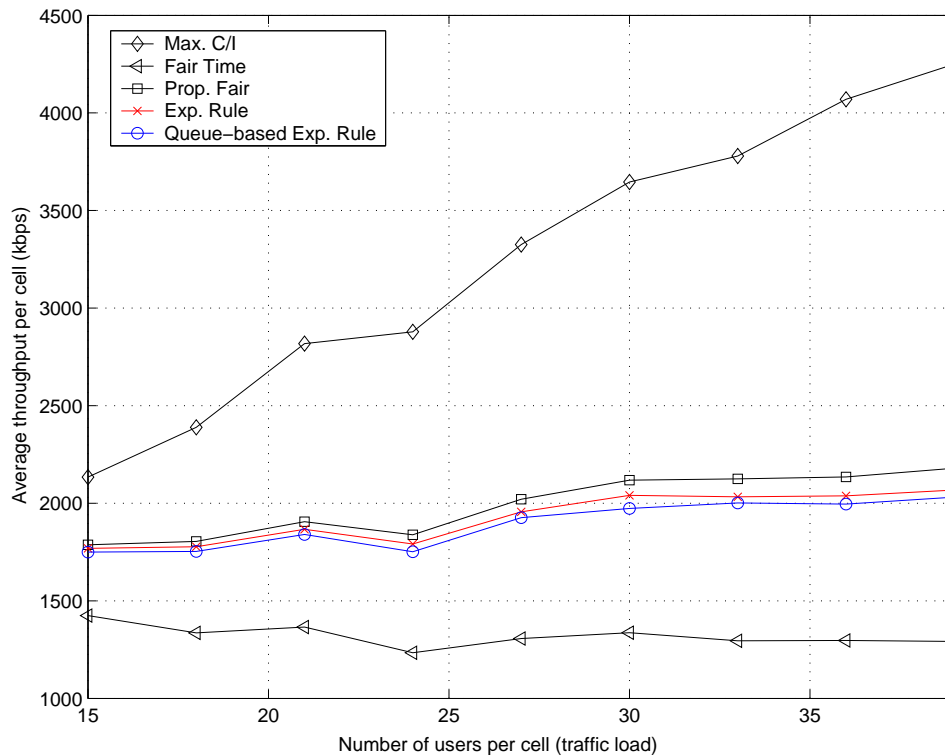


Fig. 9. System throughput performance of all schedulers with multi-code operation in a code-multiplexing fashion.

- [3] D. N. C. Tse and S. V. Hanly, "Multiaccess fading channels part I: Polymatroid structure, optimal resource allocation and throughput capacities," *IEEE Transactions on Information Theory*, vol. 44, no. 7, pp. 2796 -2815, Nov. 1998.
- [4] R. Knopp and P. A. Humblet, "Information capacity and power control in single-cell multiuser communications," *IEEE International Conference on Communications*, vol. 1, pp. 331-335, Jun. 1995.
- [5] S. Lu, V. Bharghavan, and R. Srikant, "Fair queuing in wireless packet networks," in *Proceedings of ACM SIGCOMM '97*, pp. 63-74, Cannes France, Sep. 1997.
- [6] T. S. Ng, I. Stoica, and H. Zhang, "Packet fair queueing algorithms for wireless networks with location-dependent errors," in *Proceedings of IEEE INFCOM*, pp. 1103-1111, San Francisco, CA, Mar. 1998.
- [7] S. Lu, T. Nandagopal, and V. Bharghavan, "Fair scheduling in wireless packet networks," in *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 10-20, Oct. 1998.
- [8] Y. Ofuji, A. Morimoto, S. Abeta, and M. Sawahashi, "Comparison of packet scheduling algorithms focusing on user throughput in high speed downlink packet access," *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 3, pp. 1462-1466, Sep. 2002.
- [9] A. Jalali, R. Padovani, and R. Pankaj, "Data throughput of CDMA-HDR: a high efficiency-high data rate personal communication wireless system," *Proceedings of the IEEE VTC 2000-Spring*, Tokyo, Japan, pp. 1854-1858, May. 2000.
- [10] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushyana, and S. Viterbi, "CDMA/HDR: a bandwidth efficient high speed wireless data service for nomadic users," *IEEE Communications Magazine*, vol. 38, no. 7, pp. 70-77, Jul. 2000.
- [11] S. Shakkottai and A. L. Stolyar, "Scheduling algorithms for a mixture of real-time and non-real-time data in HDR," *Bell*

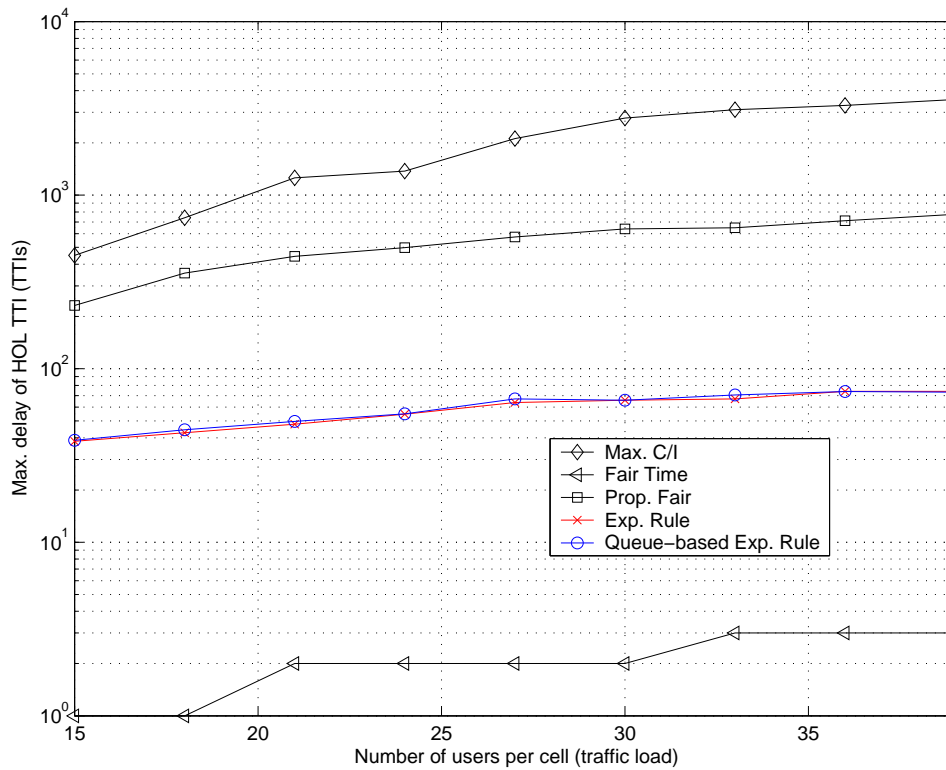


Fig. 10. The performance of maximum delay of all schedulers with multi-code operation in a code-multiplexing fashion.

Laboratories Technical Report, Oct. 2000.

- [12] S. Shakkottai and A. L. Stolyar, "Scheduling for Multiple Flows Sharing a Time-Varying Channel: The Exponential Rule," Bell Laboratories Technical Report, Oct. 2000.
- [13] A. Ekşim and M. O. Sunay, "On scheduling for delay tolerant traffic in HDR," IEEE International Symposium on Advances in Wireless Communications, pp. 189-190, Sep. 2002.
- [14] I. Stojmenovic, *Handbook of Wireless Networks and Mobile Computing*. 1st ed. John Wiley & Sons, 2002.
- [15] V. Bharghavan, L. Songwu, and T. Nandagopal, "Fair queuing in wireless networks: issues and approaches," *IEEE Personal Communications*, vol. 6, no. 1, pp. 44-53, Feb. 1999.

# Summary

## Part I

In the first part, we have defined a new performance metric --gap-processing time -- to evaluate the stall avoidance schemes for HSDPA. We derive the closed-form expressions and the probability mass functions 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 and the window size in the RLC layers, and the number of acceptable 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.

## Part II

In the second part, we have analyzed the relation among the physical layer bandwidth, the TCP layer throughput, delay and queue size when delivering the TCP traffic in the WCDMA system. The developed analytical formulas can facilitate the design of queue sizes in the buffer of base station for different TCP throughput and delay requirements. Specifically, we propose an explicit rate change notification (ERCN) mechanism to dynamically change the queue size of the buffer in the base station based on the TCP/Physical cross-layer performance issues. Hence, when the radio link capacity in the physical layer is changed, the suitable queue size in the base station can be effectively adapted and the TCP sender can accordingly change the transmit data rates, thereby both the TCP delay and throughput requirements can be met in the varying radio channel conditions. In addition, we validate the accuracy of the analysis by simulation in the EURANE, the simulation results are similar to the results of analysis. Finally, we have numerical results to show the ERCN can improve the TCP throughput, delay and delay jitter when the WCDMA system adapts the wireless link data rate to the radio channel condition.

## Part III

In the last part, we adopt a fairness index to examine the fairness performance of current link adaptation based scheduling algorithms, including the maximum C/I, fair time, proportional fair, and exponential rule schedulers. Moreover, we have proposed a new queue-based exponential rule scheduler for the HSDPA system. As summarized in Table III, the proposed scheduling algorithm outperforms all the existing scheduling algorithms in terms of fairness for the time-multiplexing fashion. In the time-multiplexing case, the proposed queue-based exponential rule scheduler improves the fairness index in a range of 15% to 50% compared to the original exponential rule algorithm, while in the code-multiplexing case the improvement reduces to the range of 5% to 15%. For the code-multiplexing configuration, Table IV compares the schedulers considered with respect to fairness, delay, and throughput. Note that although the fairness performance of the proposed queue-based exponential rule scheduler is slightly worse than the fair time scheduler in terms of the average fairness index value, the variations of the fairness index value of the proposed scheduler is smaller than the fair time scheduler, thereby still having possibility to be fairer than the fair time scheduler. It is noteworthy that in both the time-multiplexing and code-multiplexing cases, the proposed queue-based exponential rule scheduler and the original exponential scheduler improve the fairness and delay performances over the proportional fair scheduler at the cost of slightly degrading throughput. Of the comparison between code-multiplexing scheduling and time-multiplexing scheduling, we find that the time-multiplexing method is a better choice by taking all the factors of system throughput, service delay, and fairness into consideration.

# Publications

## (I) Journal

- [1] Li-Chun Wang, Chih-Wen Chang, and Chung-Ju Chang, "On the performance of Indicator-based stall avoidance for high-speed downlink access systems," submitted to IEEE Trans. on Vehicular Technology (to appear in IEEE Trans. on Vehicular Technology)
- [2] Li-Chun Wang and Chih-Wen Chang, "Gap processing time analysis for stall avoidance mechanisms for high speed downlink packet access with parallel HARQ schemes," submitted to IEEE Trans. on Mobile Computing (Mar. 2005, revised in Sep. 2005)
- [3] Li-Chun Wang, and Ming-Chi Chen, "Comparisons of Link Adaptation Based Scheduling Algorithms for the WCDMA System with High Speed Downlink Packet Access," Canadian Journal of Electrical and Computer Engineering (CJECE) vol. 29, No. 1/2, pp. 109-116, Jan./April 2004 (EI,SCI)

## (II) International Conference

- [1] Li-Chun Wang and Chih-Wen Chang, "Gap-Processing Time Analysis of an Indicator-based Stall Avoidance Mechanism," to appear in IEEE VTC Fall, Sep. 2005.
- [2] Li-Chun Wang, Chih-Wen Chang, Han-Kuang Chang, Chin-Yang Hsieh, and Sam Jiang, "Performance Comparison of Stall Avoidance Mechanisms for High Speed Downlink Packet Access in the WCDMA System," IEEE Personal, Indoor, and Mobile Radio Conference (PIMRC), Sep. 2003.
- [3] Li-Chun Wang and Chih-Wen Chang, "Gap processing time analysis for stall avoidance mechanisms for high speed downlink packet access with parallel HARQ schemes, to appear in IEEE Wirelesscomm, Symposium, June, 2005.
- [4] Li-Chun Wang and C. H. Lee, "A TCP-Physical Cross-Layer Congestion Control Mechanism for the Multirate WCDMA system Using Explicit Rate Change Notification," The First International Workshop on Information Networking and Applications, pp. 449-452, March 2005.
- [5] Li-Chun Wang and Ming-Chi Chen, "Comparisons of Link Adaptation Based Scheduling Algorithms for the WCDMA System with High Speed Downlink Packet Access," IEEE VTC, May, 2004.



## 出國報告書

撰寫時間： 94 年 10 月 2 日

姓 名	王蒞君	單 位	電信系
連絡電話	03-5712121~54511	出生年月日	53 年 3 月 19 日
職 別	副教授		
出席國際會議名稱	IEEE 62nd Semiannual Vehicular Technology Conference		
到達國家及地點	Dallas, Texas, USA		
出國期間	自 94 年 9 月 26 日 迄 94 年 9 月 28 日		
內 容	I attended the IEEE 62nd Semiannual Vehicular Technology Conference. I presented a technical paper and attend some technical lectures.  Paper Title: Li-Chun Wang and Chih-Wen Chang, "Gap-Processing Time Analysis of an Indicator-based Stall Avoidance Mechanism," to appear in IEEE VTC Fall, Sep. 2005.		

# 出國報告書

撰寫時間： 92 年 10 月 1 日

姓 名	張志文	單 位	電信系
連絡電話	03-5712121~54511	出生年月日	65 年 6 月 14 日
職 別	博班學生		
出席國際會議名稱	IEEE International Symposium on Personal, Indoor and Mobile Radio Communications		
到達國家及地點	Beijing, China		
出國期間	自 92 年 9 月 7 日迄 92 年 9 月 10 日		
內 容	<p>I attended the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications conference. I presented a technical paper and attend some technical lectures.</p> <p>Paper Title: Li-Chun Wang, Chih-Wen Chang, Han-Kuang Chang, Chin-Yang Hsieh, and Sam Jiang, "Performance Comparison of Stall Avoidance Mechanisms for High Speed Downlink Packet Access in the WCDMA System," IEEE Personal, Indoor, and Mobile Radio Conference (PIMRC), Sep. 2003.</p>		

## 出國報告書

撰寫時間： 94 年 6 月 30 日

姓名	王蒞君	單位	電信系
連絡電話	03-5712121~54511	出生年月日	53 年 3 月 19 日
職別	副教授		
出席國際會議名稱	IEEE Symposium on Wireless Protocol Engineering June 2005		
到達國家及地點	Maui, Hawaii, USA, June 2005		
出國期間	自 94 年 6 月 13 日迄 94 年 6 月 16 日		
內容	<p>I attended the IEEE Wirellesscomm, Symposium conference. I presented a technical paper and attend some technical lectures.</p> <p>Paper Title: Li-Chun Wang and Chih-Wen Chang, "Gap processing time analysis for stall avoidance mechanisms for high speed downlink packet access with parallel HARQ schemes, to appear in IEEE Wirellesscomm, Symposium, June, 2005.</p>		



# 出國報告書

撰寫時間： 93 年 5 月 30 日

姓名	王蒞君	單位	電信系
連絡電話	03-5712121~54511	出生年月日	53 年 3 月 19 日
職別	副教授		
出席國際會議名稱	IEEE 59nd Semiannual Vehicular Technology Conference		
到達國家及地點	Milan, Italy		
出國期間	自 93 年 5 月 17 日迄 93 年 5 月 19 日		
內容	<p>I attended the IEEE 59nd Semiannual Vehicular Technology Conference. I presented a technical paper and attend some technical lectures.</p> <p>Paper Title: Li-Chun Wang and Ming-Chi Chen, "Comparisons of Link Adaptation Based Scheduling Algorithms for the WCDMA System with High Speed Downlink Packet Access," IEEE VTC, May, 2004.</p>		