Dynamic Scheduling Framework on an RLC/MAC Layer for General Packet Radio Service

Jen-Shun Yang, Chien-Chao Tseng, and Ray-Guang Cheng, *Member, IEEE*

***Abstract*—In this paper, we present a traffic-scheduling framework that can dynamically allocate radio resources to a general packet radio service (GPRS) mobile station (MS) based on the interference levels of the radio links and the quality of service (QoS) specification of the MS. The underlying idea of this scheduling scheme is to preserve more bandwidth for use by those MSs that are within a low interference region so that the limited radio resources can be used more effectively. In this scheme, an MS uses a low transmission rate for data transfer when the MS is within a high interference region to avoid wasting bandwidth by transmitting data in a condition with high interference. In order to compensate for the service loss of the MS, we will allocate more bandwidth to the MS when it is within a low interference region. In addition, we also propose an analytical model that can be used to derive the transmission rate for an MS in a low interference region based on the delay-bound requirement of the MS. The performance results show that our dynamic scheme can utilize the bandwidth more effectively to satisfy various QoS requirements of the MSs in the GPRS system without changing the convolution-coding rate.**

***Index Terms*—General packet radio service (GPRS), location-dependent channel errors, quality of service (QoS), traffic scheduling.**

I. Introduction

THE packet-scheduling algorithm has been studied for many years in wired networks. However, in wireless networks, the fading characteristics of the wireless physical channel may introduce *location-dependent channel errors* and, thus, a logical connection of the radio link control (RLC) layer (i.e., RLC connection) transported on a physical channel with high fading may encounter a high bit-error ratio (BER). A high BER will normally result in retransmission traffic from the sender, and retransmission traffic causes unnecessarily long delays and wastes bandwidth. Therefore, the conventional wired packet scheduling algorithms cannot be directly applied to the wireless system, since they do not adjust the service rate of a connection according to the location-dependent BER.

Manuscript received October 30, 2001; revised April 17, 2002 and May 25, 2002; accepted June 3, 2002. The editor coordinating the review of this paper and approving it for publication is W. W. Lu. This work was supported in part by the Ministry of Education, R.O.C., under Contract 90-E-FA04-1-4 and in part by the National Science Council, Taiwan, under Contract NSC-90-2213-E-009-023.

J.-S. Yang and C.-C. Tseng are with the Department of Computer Science and Information Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: jsyang@csie.nctu.edu.tw; cctseng@csie.nctu.edu.tw).

R.-G. Cheng is with the Department of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C. (e-mail: crg@ieee.org).

Digital Object Identifier 10.1109/TWC.2003.817450

Ng *et al.* [11] have proposed a scheduling algorithm, referred to as *channel-condition independent packet fair queueing* (CIF-Q), to solve the problem of location-dependent channel errors in wireless networks by suspending transmission service of a connection of a mobile station (MS) when the MS is in a high BER region. To compensate for the service loss of the MS in a high BER region, the CIF-Q scheduling algorithm increases the service priority after the MS returns to a low BER region to fulfill its quality of service (QoS) specification. However, this suspension may cause long delays for the connection of the MS, because the duration that an MS resides within a high interference region may be unpredictably long. In many services, such as video transmissions, minor errors are acceptable to the receiver-side applications, but a long delay is not likely to be acceptable.

In this paper, we propose a scheduling scheme at the RLC/medium access control (MAC) layer of the general packet radio service (GPRS) specification [1], referring to it as a dynamic scheduling mechanism for mobile communication (DSMC). DSMC is a dynamic scheduling architecture that can conform to a variety of QoS requirements in the GPRS networks without changing the convolution-coding rate. The DSMC scheduling scheme, which is based on the self-clocked fair queuing (SCFQ) algorithm [5], can dynamically adjust the service rate (weight) for a particular connection in accordance with the channel quality. It will use the low service rate when an MS is within a high interference (high-IF) region to reduce bandwidth waste due to retransmissions, and use the high service rate when the MS is within a low interference (low-IF) region to fulfill the QoS requirements (e.g., delay bound and loss ratio) of the MS. The high and low service rates are both determined at connection setup time. Thus, the complexity of scheduling algorithm in wireless networks can be reduced.

The rest of paper is organized as follows. In Section II, we describe the DSMC architecture and the scheduling scheme. In Section III, we derive the high service rate calculation analytical model for DSMC scheduling scheme based on the interference model in a cell. Simulation results are shown in Section IV. In Section V, we discuss complexity of our scheduling algorithm. Section VI presents conclusions and future work.

II. DSMC Architecture

A. DSMC Functional Block Diagram

In GPRS, a data packet is divided into several fixed-size RLC blocks and each RLC block (henceforth referred to as "block") comprises four normal bursts transmitted on the same time

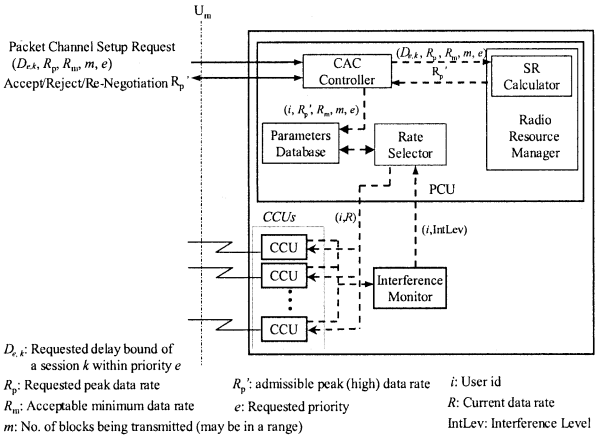


Fig. 1. Functional blocks in a BSS.

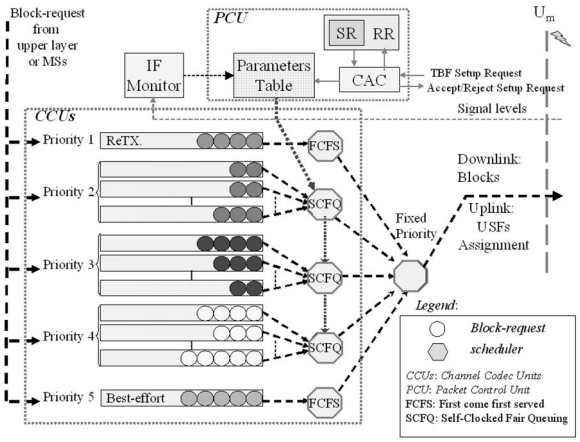


Fig. 2. Scheduling architecture in BSS.

slot in four consecutive time-division multiple access (TDMA) frames. More than one time slot of a TDMA frame can be used to fulfill the data transfer rate required for an admitted RLC connection of an MS [1], [2]. In addition, the backward error correction (BEC) mechanism is applied in GPRS, in which the receiver of a connection issues a selective automatic repeat request (ARQ) to ask the sender to retransmit an erroneous block. The waiting time of a selective ARQ request may be long and, thus, may violate the delay bound of a packet. If the delay bound of a packet is violated, then the whole packet becomes useless and the lagged blocks of this packet need to be dropped. Hence, one of the goals of the DSMC architecture is to reduce the retransmission times.

Fig. 1 shows the functional blocks of our DSMC situated at a base station. As shown in the figure, our DSMC consists of a number of channel codec units (CCUs) and a packet control unit (PCU) [2]. The functions provided by a CCU are the channel coding and the radio channel measurement, e.g., the quality–signal level received by an MS. The PCU is responsible for channel access control functions, e.g., access request–grant, as well as radio channel management functions such as congestion control. The connection admission control (CAC) controller in the PCU is responsible for the channel access control related issues. An MS may issue a connection request to the CAC controller by specifying its QoS requirements $(D_{e,k}, R_p, R_m)$, its priority level e, as well as the number of blocks m to be transmitted. $D_{e,k}$ is the delay bound required for the requested session k with a priority of e on air interface, U_m. R_p, and R_m, respectively, denote the peak and the minimum data rates requested by the session k. In other words, the MS requests that m blocks of the session k should be scheduled in a queue of priority level e, and transmitted at a minimum rate higher than R_m or a peak rate not higher than R_p before the delay bound $D_{e,k}$ is reached. We use the minimum data rate R_m as the low service rate when the MS is within a high interference region. On the contrary, the peak data rate R_p is not taken for granted as the high service rate when the MS is within a low interference region. Instead, the high service rate is determined by a simple calculation performed by the service rate (SR) calculator inside the radio resource manager. The SR calculator determines the admissible peak data rate (high service rate) R_p' that can be supported by the base station system (BSS) according to the delay bound $D_{e,k}$, and the minimum data rate R_m under a hypothetical interference model of the MS.

The interference model is based on a two-state Markov chain with transition probabilities α and β. These parameters α and β can be collected from a user behavior profile and can be updated dynamically. Details about the algorithms employed in the SR calculator are described in Section III-D. After receiving the R_p' from the SR calculator, the CAC controller can optionally accept, reject, or renegotiate with the MS. If the connection request is accepted, the CAC controller stores the parameters (R_p', R_m, m, e) to an Parameter database. The rate selector can then select and send the current data rate R to the CCU associated with the MS according to the interference level measured by the interference monitor.

The DSMC scheduling architecture can be situated in a BSS of the GPRS networks and applied to either uplink or downlink transmission. According to the specification of GPRS, block transmission is classified into five scheduling priority levels, including four data priority levels and a signal priority level, which is the highest priority level [7]. The DSMC scheduling architecture follows the specification but includes a new data priority level for the retransmission blocks. Fig. 2 shows the architecture of the DSMC scheduling algorithm. As shown in the figure, the DSMC scheduling architecture consists of five scheduling servers for data block transmission, one server for each priority level. Each server is responsible for scheduling data blocks transmitting through a single packet data channel (PDCH). The highest priority level $P1$ is for the retransmission blocks and the lowest priority level $P5$ is for the best-effort data blocks. Both $P1$ and $P5$ schedule data blocks in a first-come-first-served order. However, the scheduling servers of priority levels $P2$–$P4$ adopt the SCFQ scheduling algorithm in scheduling data blocks of QoS specific connections. In other words, multiple queues may exist in each priority of $P2$–$P4$. The queues with the same priority will be served in accordance with the SCFQ scheduling algorithm (to be explained later).

It should be noted that the block requests of a particular priority could not be served until all the blocks of the higher priority have been served. Moreover, the transmission of a block is nonpreemptable.

B. SCFQ Scheduling Algorithm

SCFQ is basically a packet-based general processor sharing scheme [3], [4] without the complex virtual clock tracking mechanism. The elimination of the virtual clock tracking mechanism from SCFQ makes SCFQ easier to implement on a high-speed network. In SCFQ, an arrival block request is tagged with a service finish time (FT) before it is placed in a queue. The service FT tag of a block request is computed from the service time and the service starting time of the block such as

$$\mathrm{FT} = \frac{\text{Block Length}}{\text{Service Rate}} + \max(\text{FT of the tail block, FT of the serving block}).$$

The service starting time of the block can be the FT of the tail block of the queue if the queue is nonempty, or otherwise, it is the FT of the serving block. Moreover, the block requests among the heads of the queues will be picked up to be served one by one in accordance with the increasing order of FT tags, and in a round-robin fashion if more than two heading blocks have the same FT tags.

III. Analytical Model for High Service Rate Calculation

In this section, we present the analytical model that is used in the SR calculator for calculating the high service rate during the setup time of a connection with QoS specification, i.e., a connection having priority level $P2$–$P4$. In order to determine the service rate, we need to first obtain the total delay of a block. The total delay can be derived from the service discipline and the input traffic. Due to the bursty nature of multimedia traffic streams, we assume that a leaky-bucket regulator is situated at the network interface of the sender site to regulate the block request flow of each session. The leaky-bucket characterization that results from the regulator will ease the analysis process because a deterministic input rate can be formulated.

In the DSMC scheduling scheme, a newly arrived block of a connection will experience three delays, including queueing delay, block transmission delay, and retransmission delay. The queueing delay is the delay that a newly arrived block waits in a queue before the block can be transmitted. Retransmission delay is the delay to retransmit an error block. It should be noted that we also have a separate retransmission queue with the highest priority in scheduling. In other words, a block is considered as "served" by the SCFQ scheduler after it is transmitted, no matter successfully or not. Moreover, a block may experience a number of attempted retransmissions before it can be transmitted successfully.

As mentioned previously, blocks of a particular priority cannot be served before all blocks with higher priorities are served, and blocks from the queues of a priority of $P2$–$P4$ are served according to the SCFQ algorithm. Therefore, a block at the head of a queue (henceforth referred to as a "heading block") may experience a priority delay, which, in turn, consists of an interpriority delay caused by all block transmissions in the higher priority levels and an intrapriority delay resulted from the SCFQ scheduling.

Let us now consider the queue that a newly arrived block enters. Each preceding block of the newly arrived block will experience a priority delay and a block transmission delay when the preceding block becomes a heading block. In addition, the newly arrived block will experience an intrapriority delay when it becomes the heading block of the queue itself. Hence, the queueing delay of a block is, thus, the summation that the priority delay and the block transmission delay experienced by all preceding blocks, plus the intrapriority delay experienced by the block. We will describe these delays in the following subsections. Some of the notations we used in the follow analytical model are summarized as follows.

L	block length (bits);
$\mathrm{BLER}_{\mathrm{hi-if}}$	block error rate when the block is transmitting within a high interference region;
$\mathrm{BLER}_{\mathrm{low-if}}$	block error rate when the block is transmitting within a low interference region;
$P_{\mathrm{Hi-if}}$	stationary probability when the MS is within a high interference region;
$P_{\mathrm{Low-if}}$	stationary probability when the MS is within a low interference region;
$(r_{e,k})^{\mathrm{hi-if}}$	low service rate of session k of priority e;
$(r_{e,k})^{\mathrm{low-if}}$	high service rate of session k of priority e;
$(\mathrm{HD}_{e,k})^{\mathrm{hi-if}}$	heading-block delay of session k of priority e within a high interference region;
$(\mathrm{HD}_{e,k})^{\mathrm{low-if}}$	heading-block delay of session k of priority e within a low interference region;
$\bar{b}_{e,k}$	mean queue length of the session k of priority e;
$\bar{w}_{e,k}$	mean queueing delay with the session k of priority e.

A. Interpriority Delay

In DSMC, we classify data block transmission into five priority classes and use a fixed priority discipline to serve these five priority classes. Therefore, the heading block selected by the server of a particular level cannot be served until all blocks of the sessions with the higher priority levels have been served. The time that the selected heading block waits for the blocks from higher priority levels to be served is called the interpriority delay.

In this subsection, we derive an upper bound for the interpriority delay. For convenience, we refer to the priority level under discussion as the priority e. Let $A_{p,s}[t_1, t_2]$ denote the amount of blocks arrived to a session s with a priority p during a time interval (t_1, t_2) and $W_{p,s}[t_1, t_2]$ is the number of blocks served for a session s with a priority p during a time interval (t_1, t_2). As shown in Fig. 3, after the $(i-1)$th selected heading block has been served, the ith heading block selected by the server of the priority e may encounter an interpriority, which is the time period (t_1', t_2). Each session j with a priority g higher than e must be served up within the time period (t_1', t_2). Thus, the amount of traffic served during the time period (t_1', t_2), $W_{g,j}[t_1', t_2]$ is equal to the amount of arrival traffic to the session during the time period (t_1, t_2), $A_{g,j}[t_1, t_2] = A_{g,j}[t_1, t_1' + \Delta t_e]$, as in (1). Here, t_1' is the epoch of the final time of serving the $(i-1)$th

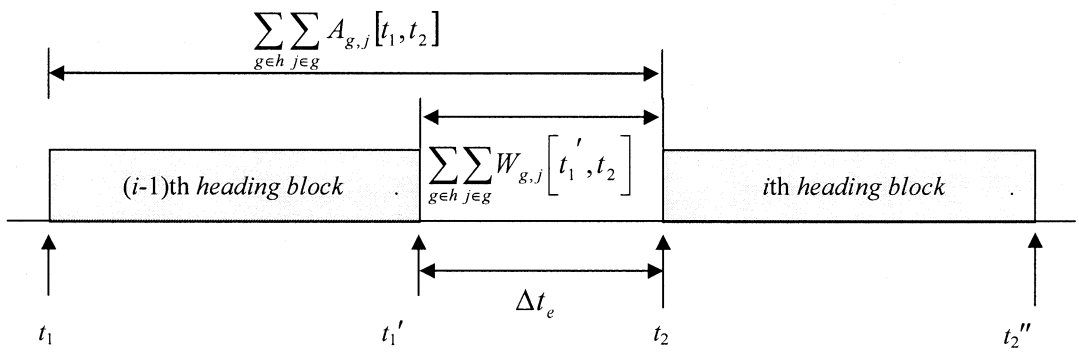


Fig. 3. Higher priority session causes delay to the particular priority.

heading block selected from the priority e and Δt_e the inter-priority delay encountered by each heading block selected from the priority e

$$W_{g,j}[t_1', t_2] = A_{g,j}[t_1, t_2], \quad \forall t_2 > t_1' > t_1, \quad t_1 \geq 0. \quad (1)$$

Let H represent the set of all priority levels higher than e. Summing up (1) for all sessions of the priorities in H, we have the following inequality:

$$\sum_{g\in H}\left[\sum_{\text{session } j\in g} W_{g,j}[t_1', t_2]\right] = \sum_{g\in H}\left[\sum_{\text{session } j\in g} A_{g,j}[t_1, t_2]\right] \quad \forall t_2 > t_1' > t_1, \quad t_1 \geq 0 \quad (2)$$

The traffic $A_{g,j}[t, t+\tau]$ during a time period $(t, t+\tau)$ has an upper bound since the DSMC scheduling architecture uses a leaky bucket to regulate the flow of each session [8]. Let $A^*_{g,j}(\tau)$ denote the upper bound of $A_{g,j}[t, t+\tau]$. We have the following inequality according to the *leaky-bucket constrained* envelope function [8]:

$$A_{g,j}[t, t+\tau] \leq A^*_{g,j}(\tau) = \sigma_{g,j} + \rho_{g,j} \cdot t \quad \forall t \geq 0, \quad \forall \tau \geq 0. \quad (3)$$

where $\sigma_{g,j}$ is the leaky-bucket size and $\rho_{g,j}$ is the token arrival rate for the session j of the priority g. Therefore, from the above leaky-bucket constraint, we can derive the following inequality:

$$\sum_{g\in H}\left[\sum_{\text{session } j\in g} A_{g,j}[t_1, t_2]\right] \leq \sum_{g\in H}\left[\sum_{\text{session } j\in g} A^*_{g,j}(t_2 - t_1)\right] \quad \forall t_2 > t_1' > t_1, \quad t_1 \geq 0 \quad (4)$$

Since the retransmission of data blocks has the highest priority and there is only one retransmission queue, we can denote the arrival traffic to the retransmission queue during the period (t_1, t_2) as $A_{1,1}[t_1, t_2]$. The retransmission traffic is the aggregated traffic of the retransmission of all sessions. By observing the system over a substantially long period of time, we can find that $A_{1,1}[t_1, t_2]$ also conforms to the leaky-bucket constraint, and we rewrite the above inequality (4) as follows:

$$A_{1,1}[t_1, t_2] + \sum_{g\in h, g\neq 1}\left[\sum_{\text{session } j\in g} A_{g,j}[t_1, t_2]\right] \leq A^*_{1,1}(t_2 - t_1) + \sum_{g\in h, g\neq 1}\left[\sum_{\text{session } j\in g} A^*_{g,j}(t_2 - t_1)\right] \quad \forall t_2 > t_1' > t_1, \quad t_1 \geq 0. \quad (5)$$

Let $P_{\text{Hi-if}}$ represent the stationary probability and $\text{BLER}_{\text{hi-if}}$ represent the block error rate that the MS is within a high-IF region; let $P_{\text{Low-if}}$ represent the stationary probability and $\text{BLER}_{\text{low-if}}$ represent the block error rate that the MS is within a low-IF region. We can derive the equation for $A^*_{1,1}(t_2 - t_1)$ as shown in the equation at the bottom of the page. By combining (2) and inequality (5), we can obtain the following inequality:

$$\sum_{g\in h}\left[\sum_{\text{session } j\in g} W_{g,j}[t_1', t_2]\right] \leq A^*_{1,1}(t_2 - t_1) + \sum_{g\in h, g\neq 1}\left[\sum_{\text{session } j\in g} A^*_{g,j}(t_2 - t_1)\right] \quad \forall t_2 > t_1' > t_1, \quad t_1 \geq 0.$$

Finally, let C_i denote the link capacity of the radio band (channel) i, and then we can obtain the following inequality by applying the inequality (3)

$$C_i \times \Delta t_e \leq A^*_{1,1}(\Delta t_e + t_1' - t_1) + \sum_{g\in h, g\neq 1}\left[\sum_{\text{session } j\in g} \sigma_{i,j} + \rho_{i,j} \times (\Delta t_e + t_1' - t_1)\right] \quad \forall t_2 > t_1' > t_1, \quad t_1 \geq 0. \quad (6)$$

From the above inequality, we can calculate the maximum value of Δt_e.

$$A_{1,1}[t_1, t_2] \leq A^*_{1,1}(t_2 - t_1) = L \times \sum_{s\in\{\text{data priorities}\}} \left\{ \left\lceil \frac{\sum_{\text{session } j\in s} \sigma_{s,j} + \rho_{s,j} \cdot (t_2 - t_1)}{L} \right\rceil \times \text{BLER}_{\text{hi-if}} \times P_{\text{Hi-if}} + \left\lceil \frac{\sum_{\text{session } j\in s} \sigma_{s,j} + \rho_{s,j} \cdot (t_2 - t_1)}{L} \right\rceil \times \text{BLER}_{\text{low-if}} \times P_{\text{Low-if}} \right\}$$

B. Intrapriority Delay

The intrapriority delay is delay contributed from the SCFQ scheduling delay. It is the delay that a heading block of a session waits for the heading blocks of some other sessions with the same priority to be served. Following the results presented by Golestani [6], we can derive the maximum intrapriority delay of the heading block of a session k with priority e as $(|\kappa_e|-1)\cdot(L/C_i)$, where C_i represents the capacity (bps) of link (band) i, κ_e represents the set of backlog sessions for priority e, and $|\kappa_e|$ represents the number of backlog sessions in the priority e.

C. Heading-Block Delay

A heading block will encounter an intrapriority delay, an interpriority delay, and a block transmission delay of its own. The heading-block delay, denoted as HD, for an MS depends on the transmission rates and can be derived as (7) or (8), respectively, for an MS within a low-IF region or within a high-IF region

$$(\mathrm{HD}_{e,k})^{\mathrm{low-if}} = |\kappa_e|\cdot\Delta t_e + (|\kappa_e|-1)\cdot\frac{L}{C_i} + \frac{L}{(r_{e,k})^{\mathrm{low-if}}} \tag{7}$$

$$(\mathrm{HD}_{e,k})^{\mathrm{hi-if}} = |\kappa_e|\cdot\Delta t_e + (|\kappa_e|-1)\cdot\frac{L}{C_i} + \frac{L}{(r_{e,k})^{\mathrm{hi-if}}} \tag{8}$$

where $(r_{e,k})^{\mathrm{low-if}}$ and $(r_{e,k})^{\mathrm{hi-if}}$ are the service rate (bps) of session k of priority e, i.e., the high and low service rate, within low-IF region and high-IF region, respectively. The item of the further right-hand side in each of the above equations represents the transmission delay of the heading block of the session k with the priority e.

Since Δt_e is likely to be brief under the control of the leaky-bucket regulator, HD is relatively small compared with the mean duration that an MS will stay in an interference region. Therefore, we have assumed that the interference condition will not change during a heading-block delay.

D. Queueing Delay

The queueing delay of a data block is the time that the block waits until the block is selected for transmission. Clearly, a data block newly arrived to a queue cannot be served by the corresponding SCFQ server until all the proceeding data blocks in the same queue have been served. Therefore, the queueing delay of a data block includes the time that the block waits for it to become a heading block itself, plus the interpriority and intrapriority delays that the block encounters when it becomes a heading block.

Let $\bar{b}_{e,k}$ represent the mean queue length of a session k with a priority e. Then from Little's result [9], the mean queue length encountered by a newly arrived data block is equal to the block arrival rate multiplied by the mean heading-block delay time. The calculation of $\bar{b}_{e,k}$ is described as

$$\bar{b}_{e,k} = \left\lceil\frac{\rho_{e,k}}{L}\right\rceil\cdot[(\mathrm{HD}_{e,k})^{\mathrm{low-if}}\cdot P_{\mathrm{Low-if}} + (\mathrm{HD}_{e,k})^{\mathrm{hi-if}}\cdot P_{\mathrm{Hi-if}}]. \tag{9}$$

As a consequence, the mean queueing delay $\bar{w}_{e,k}$ for the session k with a priority e can be calculated as

$$\bar{w}_{e,k} = \left\lceil\frac{\rho_{e,k}}{L}\right\rceil\cdot[(\mathrm{HD}_{e,k})^{\mathrm{low-if}}\cdot P_{\mathrm{Low-if}} + (\mathrm{HD}_{e,k})^{\mathrm{hi-if}}\cdot P_{\mathrm{Hi-if}}]^2 + |\kappa_e|\cdot\Delta t_e + (|\kappa_e|-1)\frac{L}{C_i}. \tag{10}$$

However, we did not use mean queueing delay directly as the queueing delay of our analysis because a data block may arrives to a session in an interference state (region) and be served in another interference state. The mean queueing delay is not precisely enough to represent the queueing delay. However, we can use the mean queueing delay to derive the probability of the interference state that an MS may stay as described in the next subsection.

1) Interference Model: We assume that the general interference model is an interrupted poisson process with transition probabilities of α and β. Hence, the duration that an MS is within the low-IF region or the high-IF region can be represented, respectively, by an exponential distribution $1-e^{-\alpha\cdot t}$, denoted as $L(t)$, or $1-e^{-\beta\cdot t}$, denoted by $H(t)$.

The interference state in which a block is served is determined by the starting state and the waiting time of the block. Therefore, we use an alternating renewal process to calculate the probability of the interference state in which a block is served. In this alternating renewal process, the block waiting time can be divided into several renewal intervals. A renewal interval consists of two exponential distributions $L(t)$ and $H(t)$. Let $F(t)$ be the convolution sum of $L(t)$ and $H(t)$. We use the notation $P_{\mathrm{LS}}^{\mathrm{hi-if}}(W)$ to represent the probability that a block with a waiting time W arrives when an MS is within a low-IF region and is served when the MS is within a high-IF region. Following the same convention, the notations of probabilities $P_{\mathrm{LS}}^{\mathrm{low-if}}(W)$, $P_{\mathrm{HS}}^{\mathrm{hi-if}}(W)$, and $P_{\mathrm{HS}}^{\mathrm{low-if}}(W)$ should be self explanatory. These four probabilities can be derived from

$$\begin{aligned}
P_{\mathrm{LS}}^{\mathrm{low-if}}(W) &= [1-L(W)] + \int_0^W [1-L(W-y)]d\left[\sum_{n=1}^{\infty}F_n(y)\right]\\
P_{\mathrm{LS}}^{\mathrm{hi-if}}(W) &= 1-P_{\mathrm{LS}}^{\mathrm{low-if}}(W)P_{\mathrm{HS}}^{\mathrm{hi-if}}(W) = [1-H(W)] + \int_0^W [1-H(W-y)]d\left[\sum_{n=1}^{\infty}F_n(y)\right]\\
P_{\mathrm{HS}}^{\mathrm{low-if}}(W) &= 1-P_{\mathrm{HS}}^{\mathrm{hi-if}}(W).
\end{aligned} \tag{11}$$

Refer to [10] for the derivation of (11).

2) Normal Delay of a Newly Arrived Block: Both the MS-terminated downlink data block and the MS-originated uplink block transmission requests may arrive at an SCFQ queue when the MS is within either a low-IF region with a probability $P_{\mathrm{low-if}}$ or a high-IF region with a probability $P_{\mathrm{hi-if}}$. Moreover, a block may be served when the MS is within a high-IF region or a low-IF region. Therefore, the normal delay of a newly arrived data block, as shown in Fig. 4, is equal to $(\bar{b}_{e,k}+1)\cdot\mathrm{HD}_{e,k}$, where $\mathrm{HD}_{e,k}$ can be $(\mathrm{HD}_{e,k})^{\mathrm{low-if}}$, or $(\mathrm{HD}_{e,k})^{\mathrm{hi-if}}$. By using the above (11), we can obtain the

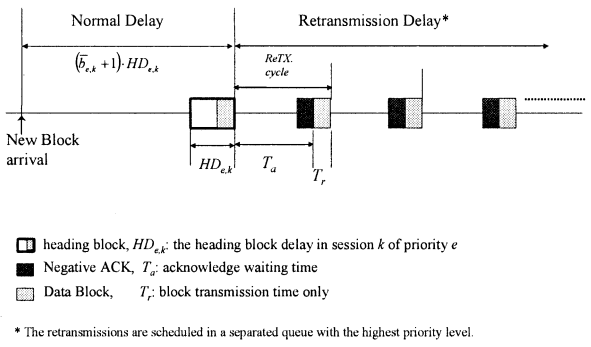


Fig. 4. Transmission of a block.

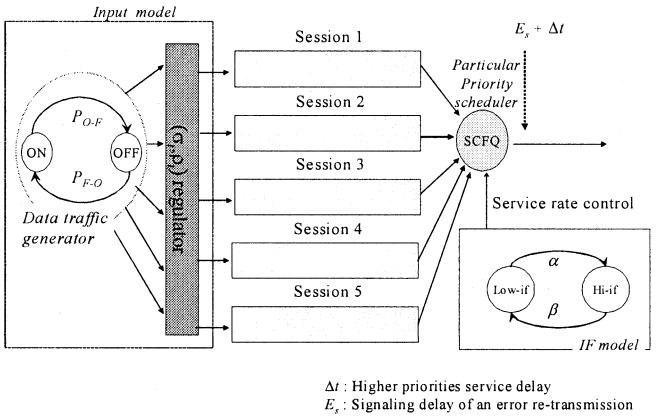


Fig. 5. Simulation architecture.

normal delay $(\mathrm{ND}_{e,k})$ of a block with a mean waiting time of $\bar{w}_{e,k}$ as follows:

$$\begin{aligned}\mathrm{ND}_{e,k} &= P_{\mathrm{Low-if}}\cdot\left[(\bar{b}_{e,k}+1)\cdot(\mathrm{HD}_{e,k})^{\mathrm{low-if}}\cdot P_{\mathrm{LS}}^{\mathrm{low-if}}(\bar{w}_{e,k})\right.\\ &\quad\left.+(\bar{b}_{e,k}+1)\cdot(\mathrm{HD}_{e,k})^{\mathrm{hi-if}}\cdot P_{\mathrm{LS}}^{\mathrm{hi-if}}(\bar{w}_{e,k})\right]\\ &\quad+P_{\mathrm{Hi-if}}\cdot\left[(\bar{b}_{e,k}+1)\cdot(\mathrm{HD}_{e,k})^{\mathrm{hi-if}}\cdot P_{\mathrm{HS}}^{\mathrm{hi-if}}(\bar{w}_{e,k})\right.\\ &\quad\left.+(\bar{b}_{e,k}+1)\cdot(\mathrm{HD}_{e,k})^{\mathrm{low-if}}\cdot P_{\mathrm{HS}}^{\mathrm{low-if}}(\bar{w}_{e,k})\right].\end{aligned}\tag{12}$$

E. Retransmission Delay

A data block may need to be retransmitted several times before it can be transmitted successfully. Since the BEC mechanism is adopted in GPRS, each retransmission consists of two delays, the waiting time T_a of a selective ARQ request, and the retransmission time T_r, as shown in Fig. 4. The transmission of a data block and the retransmission of this block may occur in either interference condition. In other words, a retransmission cycle may start and end in either interference region. Let A denote the mean retransmission delay starting from a low-IF region, whereas B denotes the mean retransmission delay starting from a high-IF region. Hence, we can describe the mean retransmission delays A and B by two cross *recursive* equations, as shown below. As a consequence, the retransmission delay $(\mathrm{RD}_{e,k})$ of a block can be obtained as

$$\begin{aligned}A &= \mathrm{BLER}_{\mathrm{low-if}}\left\{(T_a+T_r)+A\cdot P_{\mathrm{LS}}^{\mathrm{low-if}}(T_a+T_r)\right.\\ &\quad\left.+B\cdot P_{\mathrm{LS}}^{\mathrm{hi-if}}(T_a+T_r)\right\}\\ B &= \mathrm{BLER}_{\mathrm{hi-if}}\left\{(T_a+T_r)+B\cdot P_{\mathrm{HS}}^{\mathrm{hi-if}}(T_a+T_r)\right.\\ &\quad\left.+A\cdot P_{\mathrm{HS}}^{\mathrm{low-if}}(T_a+T_r)\right\}\\ \mathrm{RD}_{e,k} &= P_{\mathrm{Low-if}}\cdot\left\{A\cdot P_{\mathrm{LS}}^{\mathrm{low-if}}\left(\bar{w}_{e,k}+\frac{L}{(r_{e,k})^{\mathrm{low-if}}}\right)\right.\\ &\quad\left.+B\cdot P_{\mathrm{LS}}^{\mathrm{hi-if}}\left(\bar{w}_{e,k}+\frac{L}{(r_{e,k})^{\mathrm{hi-if}}}\right)\right\}\\ &\quad+P_{\mathrm{Hi-if}}\cdot\left\{B\cdot P_{\mathrm{HS}}^{\mathrm{hi-if}}\left(\bar{w}_{e,k}+\frac{L}{(r_{e,k})^{\mathrm{hi-if}}}\right)\right.\\ &\quad\left.+A\cdot P_{\mathrm{HS}}^{\mathrm{low-if}}\left(\bar{w}_{e,k}+\frac{L}{(r_{e,k})^{\mathrm{low-if}}}\right)\right\}.\end{aligned}\tag{13}$$

1) High Service Rate of a Session: By summing up the normal delay (12) and the retransmission delay (13) of a newly arrived block, the SR calculator can calculate the total delay $(\mathrm{TD}_{e,k})$ for a newly arrived block in a session k with a priority e. To simplify the calculation of the high service rate of a session, we assume that the low service rate of a session when the MS is within a high-IF region to be a predefined value. The value of this low service rate can be chosen by a user application in accordance with the required characteristics of the media stream used in the application. For example, the low service rate can be assigned as the minimum tolerable decoding rate of a Motion Pictures Expert Group (MPEG) video stream.

If we assume that a session k with a priority e has a QoS specification for m block transmissions with air-interface delay bound $D_{e,k}$, then the SR calculator can calculate the high service rate of the session k with a priority e under the constraint of the inequality

$$\mathrm{TD}_{e,k}\le D_{e,k}/m.\tag{14}$$

IV. Simulation Results

In this section, we present the simulation we conducted to evaluate the performance result of our proposed method. The architecture of the simulation is shown in Fig. 5. Instead of showing the sessions in all priority levels, we focus our discussion only on a particular priority level. In Fig. 5, we assume that there are five active sessions in the particular priority level under discussion. The input model of our simulation consists of two concatenated parts: the data-traffic generation part and the regulator part. A two-state (ON and OFF) interrupted Bernoulli process model with parameters P_{O-F} and P_{F-O} is used to generate the input data stream in the data-traffic generation part. The regulator acts as a leaky-bucket policer with parameters (σ,ρ), where σ is the leaky-bucket size in bits and ρ is the token arrival rate in blocks-per-timeslot (bpt), to regulate the input traffic smoothly to each session. Without loss of generality, we measure the simulation results for the sessions of the priority level four in our simulations, that is, there are three higher priority levels existing in the DSMC schedulers, one for retransmission and two for data priority level's transmission. The number of active (ON) sessions in each of the two higher data priority levels

follows the Binomial distribution. For the two higher data priority levels, σ and ρ of the leaky-bucket policers are all fixed at 800 b and 1600 bpt, respectively. Furthermore, the block size is fixed at 500 b, close to the size of a block in GPRS.

The interference model (IF-model) in our simulation model is based on a two-state Markov chain with interference state-transition probabilities α (low to high) and β (high to low). We assume that the interference state-transition probability $\alpha = 0.1$ and $\beta = 0.9$ within a cell. The scheduler periodically checks the interference state associated with an MS to decide whether the high or low service rate should be used for a session of the MS. Without loss of generality, we assume that the average BERs in high- and low-IF regions are 10^{-3} and 10^{-5}, respectively, in a GPRS cell.

We compare the performance results of three scheduling schemes, the SCFQ scheduling scheme, the CIF-Q scheduling scheme, and our DSMC scheduling scheme, in terms of mean delay, mean number of retransmissions, block dropping ratio, and remaining capacity. In the simulation, the arrival rate of the input traffic to a session could be adjusted by varying the arrival peak rate and the leaky rate ρ. Normally, the performance difference of the scheduling schemes is not clear in a system under light load. Therefore, we use high peak and leaky rates to obtain the arrival rates of a session to observe the performance results when the system is heavily loaded.

In the simulation, we generate a new connection service request once a session is served and run the simulations for a long-time to obtain the steady-state results. The low service rate of the DSMC scheduling scheme is set to 500 bpt for all sessions. The high service rate for a session is computed according to the inequality (14) when a new session service request of the session is being issued. For the SCFQ scheduling scheme, the service rate for each connection is fixed to a value determined, when the request of the connection is being issued, from the delay bound and the amount of data to be transferred for the connection. Similar to the SCFQ scheme, the CIF-Q scheduling scheme uses an initial service rate for a connection when the connection is being established. However, instead of using a fixed service rate for all blocks of the connection, the CIF-Q scheduling scheme suspends the service of the connection when the connection is in a high-IF state, and increases the service priority of the connection when the connection returns to a low-IF state.

Figs. 6–9 show the comparison of the above three scheduling schemes in terms of mean delay, mean number of retransmissions, block-dropping ratio, and remaining bandwidth capacity, respectively. The X-axis in each figure represents the mean arrival rate of the input traffic per session in the priority level four. Fig. 6 shows that the SCFQ scheduling scheme has the worst mean delays among the three scheduling schemes. This long delay of the SCFQ scheme is due to the excessive number of retransmissions, as can be seen in Fig. 7. Fig. 7 also shows that the mean number of retransmissions for the CIF-Q scheduling scheme is slightly less than our DSMC scheduling scheme. This is because the CIF-Q scheduling scheme suspends the service when an MS is within a high-IF region, and thus, there are no retransmissions in the high-IF regions. However, as mentioned before, this suspension may cause long delays for the data transmission because the duration of a high-IF region may be unpredictably long. Therefore, we can observe from Fig. 6 that the DSMC scheduling scheme has lower delays than the CIF-Q scheduling scheme. We may conclude that our DSMC scheduling scheme is more robust in terms of guaranteeing shorter delays.

Fig. 8 shows the comparison of the block-dropping ratio among three schemes. In the simulation, a whole block is

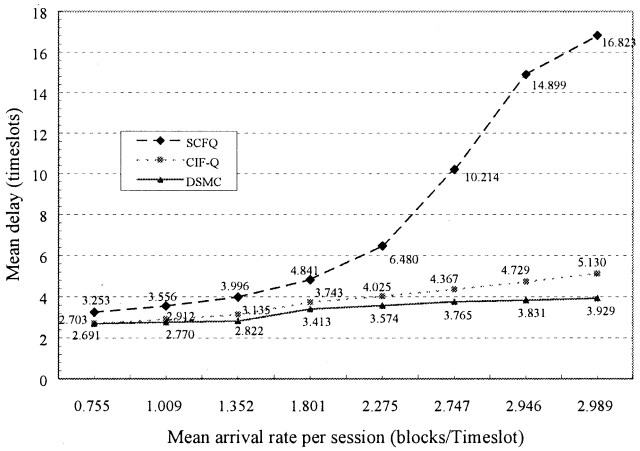


Fig. 6. Comparison of mean delay.

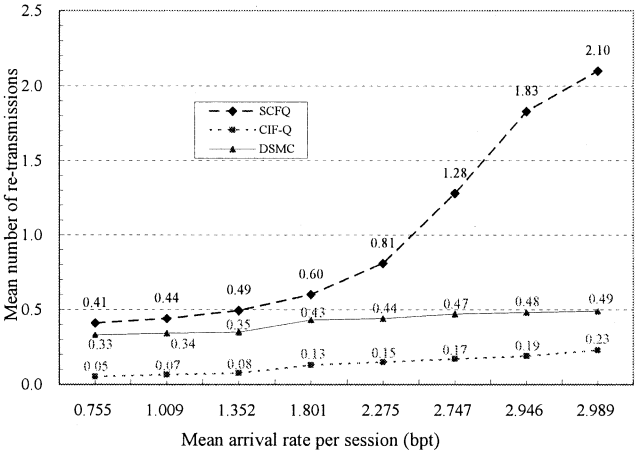


Fig. 7. Comparison of mean number of retransmissions.

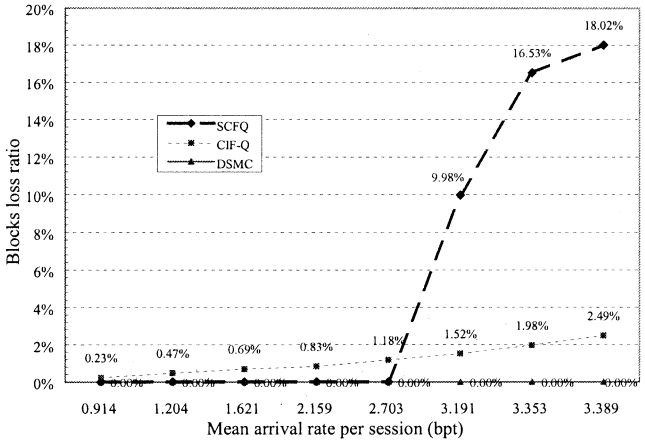


Fig. 8. Comparison of block dropping ratio.

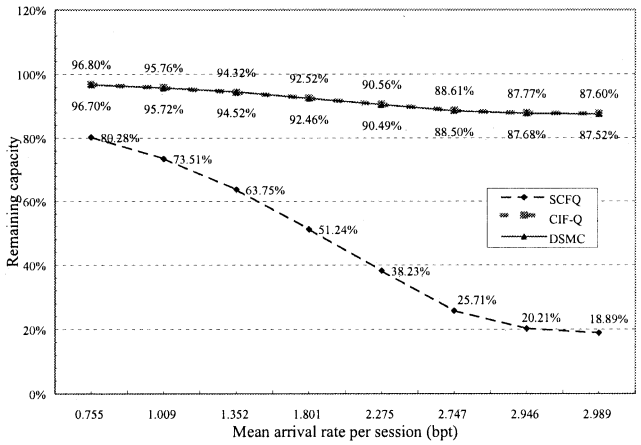


Fig. 9. Comparison of remaining bandwidth capacity.

dropped if the block cannot be transmitted in time to fulfill the delay bound. The curve of the SCFQ scheduling scheme rises up at 2.703 bpt. This phenomenon is because the system is saturated by the retransmission and high priority data transmission around this point and the data blocks of the sessions with priority level four cannot be transmitted in time to fulfill the delay bound. Similarly, the block-dropping ratio in the CIF-Q scheduling scheme is higher than the DSMC scheduling scheme since the CIF-Q scheme suspends data transmission of an MS when the MS is within a high-IF region, thus, causing the delay bound to be violated. We may conclude that our DSMC scheduling scheme outperforms the other two schemes in terms of guaranteeing limited losses. Moreover, since the numbers of retransmissions have been reduced in DSMC scheme, it utilizes the system bandwidth more efficiently.

Fig. 9 shows the comparison of the remaining bandwidth capacity among these three schemes under various arrival rates of input traffic. From the figure, we can find that the curves of CIF-Q and DSMC are almost identical. Both the CIF-Q and DSMC scheduling schemes still have over 80% bandwidth capacity left at 2.989 bpt, whereas the SCFQ scheduling scheme is overloaded if the arrival rate is over 2.7 bpt. This is because both the CIF-Q and DSMC scheduling schemes do not waste bandwidth in inefficient retransmissions when an MS is within high-IF regions. Therefore, the CIF-Q and the DSMC scheduling schemes can serve more users than the SCFQ in a system with the same bandwidth capacity. However, the time complexity of the CIF-Q scheduling scheme is higher than that of the DSMC scheduling scheme. We discuss the algorithm complexity in Section V.

V. Algorithm Complexity

The time complexity of the SCFQ scheduling scheme is the time complexity in selecting the highest weighted block among the heads of the queues. This selection can be done in $O(\log n)$, where n is the number of active sessions in the priority level under discussion. The selection of the highest weighted block in our DSMC scheduling scheme is the same as that in the SCFQ scheduling scheme. However, our DSMC scheduling scheme needs to dynamically adjust the weights of the heading block of each active session according to the interference state of the active sessions. Therefore, the time complexity of our DSMC scheduling scheme is dominated by the above weight-adjusting process, and has a time complexity of $O(n)$. The algorithm complexity of the CIF-Q scheduling scheme is $O(n \log n)$, and can be found in [11].

VI. Conclusion and Future Work

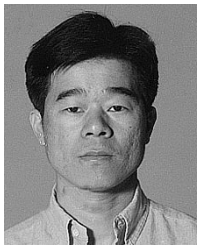
In this paper, we have proposed a DSMC scheduling architecture that can adjust the service rate of a connection according to the interference condition of the connection. The DSMC architecture, instead of changing the convolution-coding rate, has the advantage of avoiding unnecessary retransmissions that may occur when an MS is within a high-IF region. It should be noted that the DSMC scheduling scheme determines the service rate of a connection at the setup time of the connection. From the simulation results, we find that the DSMC scheduling scheme not only can guarantee the delay bound and block-dropping ratio, but also can utilize the bandwidth more efficiently. However, further research is need to determine whether the high service rate could be calculated according to the behavior profile of MSs, since the interference state changes are highly dependent on the behaviors of MSs.

References

[1] J. Cai and D. Goodman, "General packet radio service in GSM," *IEEE Commun. Mag.*, vol. 35, pp. 122–131, Oct. 1997.
[2] *GSM 03.64 Overall Description of the GPRS Radio Interface, Stage 2*, ETSI Standard v.7.0.0, 1999.
[3] W. Stallings, *High-Speed Networks: TCP/IP and ATM Design Principles*. Englewood Cliffs, NJ: Prentice-Hall, 1998, pp. 325–330.
[4] A. Demers, S. Keshav, and S. Shenkar, "Analysis and simulation of a fair queuing algorithm," in *Proc. SIGCOMM'89*, Austin, TX, Sept. 1989, pp. 1–12.
[5] S. J. Golestani, "A self-clocked fair queuing scheme for broadband applications," in *Proc. IEEE INFOCOM'94*, Apr. 1994, pp. 636–646.
[6] ——, "Network delay analysis of a class of fair queuing algorithms," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 1057–1070, Aug. 1995.
[7] *GSM 03.60 General Packet Radio Service (GPRS); Service Description; Stage 2*, ETSI Standard v.7.0.0, 1999.
[8] R. L. Cruz, "A calculus for network delay—Part I: Network elements in isolation," *IEEE Trans. Inform. Theory*, vol. 37, pp. 114–131, Jan. 1991.
[9] L. Kleinrock, *Queueing Systems Volume I Theory*. New York: Wiley, Mar. 1974, pp. 17–18.
[10] S. Ross, *Stochastic Processes*, 2nd ed. New York: Wiley, Mar. 1980, pp. 114–115.
[11] T. S. E. Ng, I. Stoica, and H. Zhang, "Packet fair queueing algorithms for wireless networks with location-dependent errors," in *Proc. INFOCOM'98*, vol. 3, 1998, pp. 1103–1111.

Jen-Shun Yang received the B.S. degree in electronic engineering from the Kuang-Wu Institute of Technology, Taiwan, R.O.C., in 1987 and the M.Sc. degree in computer engineering and science from the Yuan-Ze University, Taiwan, R.O.C., in 1995. He is currently working toward the Ph.D. degree at the National Chiao-Tung University, Hsinchu, Taiwan, R.O.C.

From September 1995 to September 1996, he was a Research Assistant in the Communication and Multimedia Laboratories, Yuan-Ze University, Taiwan, R.O.C. His research interests include high-speed networks, wireless communication, and wireless/wireline networks integration.



Chien-Chao Tseng received the B.S. degree in industrial engineering from the National Tsing-Hua University, Hsinchu, Taiwan, R.O.C., in 1981, and the M.S. and Ph.D. degrees in computer science from the Southern Methodist University, Dallas, TX, in 1986 and 1989, respectively.

He is currently a Professor in the Department of Computer Science and Information Engineering at the National Chiao-Tung University, Hsinchu, Taiwan, R.O.C. His research interests include mobile computing, and wireless internet.

Ray-Guang Cheng (S'94–M'97) was born in Taiwan, R.O.C. He received the B.E., M.E., and Ph.D. degrees in communication engineering from the National Chiao-Tung University, Hsinchu, Taiwan, R.O.C., in 1991, 1993, and 1996, respectively.

From 1997 to 2000, he was with Advance Technology Center, Computer and Communication Laboratories, Industrial Technology Research Institute as a Researcher. He was involved in the designing of MAC and radio resource management (RRM) algorithms for GPRS/W-CDMA systems. From 2000 to 2003, he was with BenQ Mobile System Inc., Hsinchu, Taiwan, R.O.C., as a Manager of the R&D Division and is involved in the Third-Generation UMTS Terrestrial Radio Access Network (UTRAN) Project. In 2003, he joined the Department of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C., as an Assistant Professor. His research interests include RRM algorithms of GPRS/W-CDMA systems and asynchronous transfer mode (ATM) networks.

Dr. Cheng has been a member of the Phi Tau Phi Scholastic Honor Society since 1993. He received the Best Industrial-Based Paper Award from the Ministry of Education, Taiwan, in 1998. He led the 3G Protocol Project and his team was named Top Research Team of the Year by ITRI, in 2000. His team also received the Outstanding Technology Prize from the Ministry of Economic Affairs, Taiwan, in 2000.