Time-Division-Based Cyclic Scheduling for UMTS High-Speed Downlink Shared-Channels

Chai-Hien Gan, *Member, IEEE*, Nei-Chiung Perng, Phone Lin, *Senior Member, IEEE*, and Tei-Wei Kuo, *Senior Member, IEEE*

***Abstract*—Third Generation Partnership Project working group proposes the high-speed downlink packet access to speed up downlink transmission for the universal mobile telecommunication system, where the time-division-based high-speed downlink shared-channel (HS-DSCH) approach is adopted. Our previous work proposed the shared-channel assignment and scheduling (SCAS) scheme to efficiently schedule the DSCH to serve different connections, which guarantees the requested transmission rate for each connection. However, limitations exist in the SCAS scheme: 1) the requested transmission rates of connections must be two's power numbers of the basic transmission rate and 2) all served connections should be rescheduled while a new request is granted, which introduces extra rescheduling overhead to the system. This paper proposes the elastic shared-channel assignment and scheduling (eSCAS) scheme to overcome these limitations. We formally prove the correctness of the eSCAS scheme. An analytical model and simulation experiments are conducted to compare the performance for eSCAS and SCAS. Our study shows that eSCAS can significantly improve the acceptance rate for new connections without increasing the rescheduling overhead.**

***Index Terms*—High-speed downlink packet access (HSDPA), high-speed downlink shared-channel (HS-DSCH), orthogonal variable spreading factor (OVSF), scheduling, universal mobile telecommunication system (UMTS).**

Manuscript received June 19, 2006; revised September 8, 2006, October 5, 2006, and October 13, 2006. The work of P. Lin was supported in part by the National Science Council (NSC), R.O.C., under Contracts NSC 94-2627-E-002-001- and NSC 95-2221-E-002-091-MY3, by the Ministry of Economic Affairs (MOEA), R.O.C., under Contract 93-EC-17-A-05-S1-0017, by the Chunghwa Telecom Laboratories, Telcordia Applied Research Center, Taiwan Network Information Center (TWNIC), and by the Iventec Appliances Corporation, and by the Excellent Research Projects of National Taiwan University under 95R0062-AE00-07. The work of C.-H. Gan was supported in part by the NSC, R.O.C., under Contracts NSC 95-2119-E-009-010, NSC 95-2119-E-009-019, and NSC 95-2218-E-009-201-MY3, by the Chunghwa Telecom Laboratories, Taiwan Network Information Center (TWNIC), and by the ICL/ITRI, Taiwan, R.O.C. The work of T.-W. Kuo was supported in part by a research grant from the NSC under Grants NSC 93-2752-E-002-008-PAE and 94-2213-E-002-007 and in part by a research grant from the Academia Sinica. The review of this paper was coordinated by Dr. J. Misic.

C.-H. Gan is with the Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: chgan@csie.nctu.edu.tw).

N.-C. Perng is with the Genesyslogic, Inc., Taipei 231, Taiwan, R.O.C. (e-mail: Neil.Perng@genesyslogic.com.tw).

P. Lin is with the Department of Computer Science and Information Engineering, the Graduate Institute of Networking and Multimedia, and the College of Electrical Engineering and Computer Science, National Taiwan University, Taipei 106, Taiwan, R.O.C. (e-mail: plin@csie.ntu.edu.tw).

T.-W. Kuo is with the Department of Computer Science and Information Engineering, the Graduate Institute of Networking and Multimedia, and also with the College of Electrical Engineering and Computer Science, National Taiwan University, Taipei 106, Taiwan, R.O.C. (e-mail: ktw@csie.ntu.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2007.897648

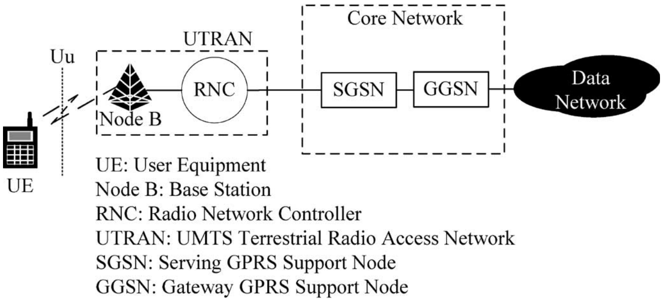

Core Network
UTRAN
Uu
RNC
SGSN
GGSN
Data Network
Node B
UE
UE: User Equipment
Node B: Base Station
RNC: Radio Network Controller
UTRAN: UMTS Terrestrial Radio Access Network
SGSN: Serving GPRS Support Node
GGSN: Gateway GPRS Support Node

Fig. 1. UMTS network architecture.

I. Introduction

The universal mobile telecommunication system (UMTS) [1] proposed by Third Generation Partnership Project (3GPP) provides high-speed wireless transmission service up to 2 Mb/s. Fig. 1 shows the UMTS network architecture consisting the UMTS terrestrial radio access network (UTRAN) and the core network. The UTRAN includes radio base stations (Node Bs) and network radio controllers (RNCs). User equipment (UE) access the UTRAN through the wide-band code division multiple access (WCDMA)-based air interface Uu [2]. The core network routes packets for users.

In WCDMA, the orthogonal variable spreading factor (OVSF) code is used to preserve the orthogonality among the radio channels for different users, where user data are encoded with the OVSF code before being sent. The length of the code sequence is known as spreading factor (SF). By assigning OVSF codes with different SFs, variable transmission rates can be achieved [3], [4]. The OVSF codes are generated based on a K-layer complete binary tree, as shown in Fig. 2. Each node in the tree corresponds to an OVSF code. An OVSF code is denoted as $C_{k,n}$, where k $(0 \leq k \leq K)$ and n $(1 \leq n \leq 2^k)$ are the layer number and the sequence number, respectively. The SF of a code in layer k is 2^k. Assume that the radio channel assigned the OVSF code $C_{K,n}$ provides the transmission rate r (in bits per second). Then, the radio channel with the OVSF code $C_{k,n}$ can support the transmission rate $R_{k,n} = 2^{K-k}r$ (in bits per second). Typically, the r value is 8 × 1024. The r value may vary according to several physical layer factors including signal-to-noise ratio, bit error rate, total transmission power, etc. [3]. The discussion of these factors is out of the scope of this paper, which is not included in this paper. We focus on how to schedule an OVSF code to serve different connections according to the requested transmission rate of a connection, i.e., packet scheduling.

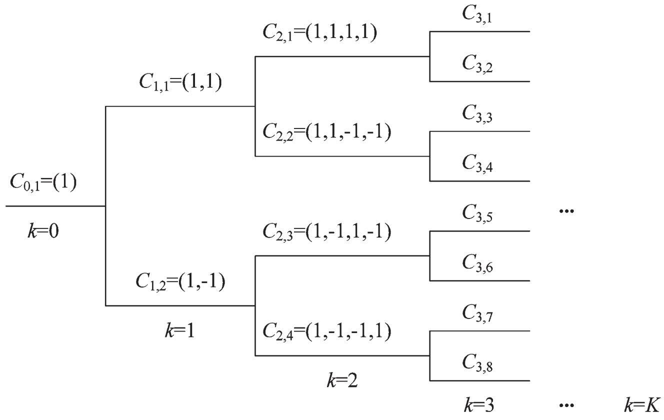


Fig. 2. OVSF code tree.

Four types of applications (including conversational, multimedia, interactive, and background applications) are identified in UMTS. The existing UMTS system can provide transmission services that satisfy most requirements of these applications. However, for the advanced multimedia applications, higher transmission rate service is required. For example, to run video-on-demand applications smoothly (where MPEG-1 video files are delivered), it requires transmission rate 300 kb/s. To speed up the downlink transmission, 3GPP proposed the high-speed downlink packet access (HSDPA) technology [5], [6]. The key technologies of HSDPA include link adaption, fast physical layer (L1) retransmission combining, high-speed downlink shared-channel (HS-DSCH), and the movement of the retransmission procedure from RNC to Node B. In this paper, we converge our discussion on scheduling for the HS-DSCH.

The HS-DSCH is a logical channel with which different users share the same OVSF code channel in a time-division manner. A high-speed OVSF code channel is dedicated to one user within a moment of time [which is known as a transmission time interval (TTI)] so that the user data can be delivered with high transmission rate in a short time period [6]. Scheduling an HS-DSCH to serve different connections is an important issue, which significantly affects the utilization of the HS-DSCH. Several studies have touched on this issue, which are summarized as follows.

Fattah and Leung [7] provided an overview of scheduling techniques in wireless multimedia networks. Parekh and Gallager [8] proposed an idealized fluid flow algorithm, i.e., generalized processor sharing (GPS) that serves all connections simultaneously. One assumption in GPS is that time can be infinitely divisible. This assumption is not practical. Many researches including weighted fair queueing [9], virtual clock [10], and weighted round-robin [11] algorithms focused on the emulation of the GPS algorithm. However, these algorithms introduced high computation complexity. In the HSDPA technology, the shorter response time from a UE to the scheduler is required since the scheduler is moved from RNC to Node B. Furthermore, these algorithms have much complexity on scheduling, which are not easily implemented in the real system. In our previous research [12], [13], we proposed the *shared-channel assignment and scheduling* (SCAS) scheme to schedule DSCHs. SCAS has been shown taking less complexity than other existing scheduling schemes do [13]. The SCAS scheme is briefly described as follows.

When a request Q_n requesting transmission rate $\Gamma_n = 2^{a_i}r$ (in bits per second) arrives, the SCAS scheme first finds a shared channel $C_{k,n}$ supporting transmission rate $R_{k,n}$ that satisfies $R_{k,n} - \sum_{Q_i \in S_{k,n}} \Gamma_i \geq \Gamma_n$, where $S_{k,n}$ contains the connections served by the channel $C_{k,n}$. To serve Q_n using $C_{k,n}$, the SCAS scheme "reschedules" all connections in the $S_{k,n}$ set by executing the following steps. For all $Q_i \in S_{k,n}$, the SCAS scheme determines the *periodic value* $P_i = D(R_{k,n}/\Gamma_i)$, where D is the length of a TTI. The SCAS scheme reserves TTIs for the connections with smaller P_i's first and reserves in any order if tie breaking occurs (i.e., more than one connection have the same P_i value). The reservation rule is that if the TTI beginning at the *initial time* O_i is reserved for Q_i, the next TTI reserved for Q_i must be apart from O_i for P_i (i.e., at $O_i + P_i$). The SCAS scheme has the following two limitations.

- First, the transmission rate Γ_i requested by each downlink connection Q_i must be a two's power number of r (in bits per second) (i.e., $2^{a_i}r$ b/s), which is referred to as *two's power limitation*. For example, to accommodate a Q_i whose requested transmission rate is not a two's power number of r (in bits per second), the network should reserve higher transmission rate for Q_i (i.e., Q_i is served with transmission rate higher than the requested transmission rate).
- Second, all the connections served by the HS-DSCH must be rescheduled to accommodate Q_i in the HS-DSCH, which is referred to as *rescheduling overhead*.

In this paper, we propose the *elastic shared-channel assignment and scheduling* (eSCAS) scheme to release the aforementioned two limitations.

The rest of this paper is organized as follows. Section II presents the eSCAS scheme. Section III proves the correctness of eSCAS. In Section IV, we conduct analysis and simulation experiments to compare the performance for SCAS and eSCAS. The concluding remark is given in Section V.

II. eSCAS Scheme

This section describes the eSCAS scheme. For the sake of clarity, we refer the connections requesting transmission rate $\Gamma_i = 2^{a_i}r$ (in bits per second), where $a_i \in \mathcal{N}$ as *two's power connections*, and the others [whose requesting transmission rates are not two's power numbers of r (in bits per second)] as *arbitrary connections*. Note that as mentioned previously, the r value may be different for each connection. The connection will request the transmission rate $a \times r$ (in bits per second) under current network physical layer status, e.g., signal-to-noise ratio and transmission power. If the r value changes due to the change of the network status, the connection may rerequest the new transmission rate $a' \times r$ (in bits per second). The eSCAS scheme first divides an arbitrary connection into a set of two's power subconnections, and then uses the SCAS scheme to schedule these subconnections. For more advancement, the eSCAS scheme attempts to reduce the rescheduling overhead

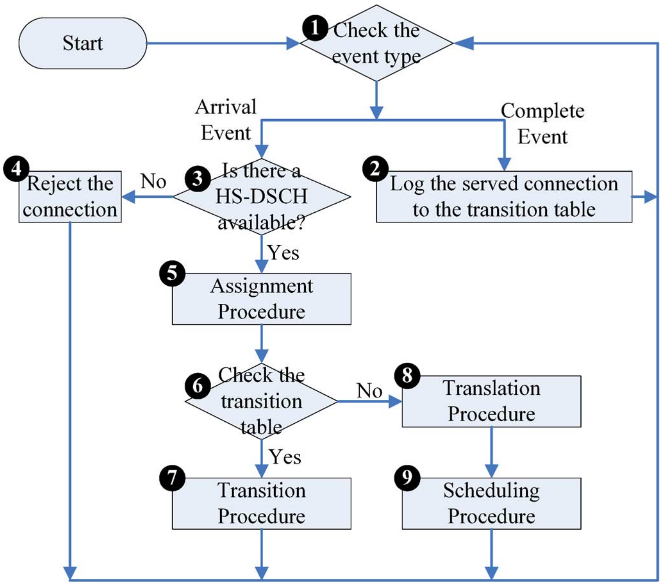


Fig. 3. Flowchart for the eSCAS scheme.

in the previously proposed SCAS scheme. The eSCAS scheme consists of four procedures: 1) assignment; 2) translation; 3) scheduling; and 4) transition. The notation used in this scheme is listed as follows.

- $C_{k,n} = (R_{k,n})$: The HS-DSCH with the OVSF code $C_{k,n}$ and total transmission rate $R_{k,n} = 2^{K-k}r$ (in bits per second).
- $S_{k,n}$: The set containing all connections served by the same HS-DSCH $C_{k,n}$.
- $\Pi_{k,n}$: The transition table of the HS-DSCH $C_{k,n}$.
- $Q_i = (\Gamma_i)$: The ith connection in $S_{k,n}$, which requests the transmission rate Γ_i (in bits per second).
- $q_{i,j} = (\gamma_{i,j})$: The jth two's power subconnection of the connection Q_i, which requests the transmission rate $\gamma_{i,j}$ (in bits per second).
- P_i and O_i: The periodic value and initial time for the connection Q_i, respectively.
- $p_{i,j}$ and $o_{i,j}$: The periodic value and initial time for the subconnection $q_{i,j}$, respectively.
- D: The length of a TTI.

Fig. 3 illustrates the flowchart of the eSCAS scheme. Two types of events are defined for a connection, which are the *connection arrival event* and the *transmission complete event*. We introduce a *transition table* $\Pi_{k,n}$ for the HS-DSCH $C_{k,n}$. Each entry in $\Pi_{k,n}$ consists of three kinds of information: 1) the requested transmission rate Γ_i; 2) the initial time O_i; and 3) the periodic value P_i for the connection Q_i (that is previously served by the HS-DSCH $C_{k,n}$). Upon receipt of an event, the eSCAS scheme checks the type of the event (see 1 in Fig. 3). If the event is a transmission complete event, the eSCAS scheme logs the configuration information of the TTIs (reserved for this completed connection) into $\Pi_{k,n}$ (see 2 in Fig. 3). The configuration information will be referenced to serve a new connection (to be elaborated later), and thus the rescheduling overhead may be reduced. If the event is a connection arrival event, the eSCAS scheme checks whether there is a suitable HS-DSCH that can serve the new connection (see 3 in Fig. 3).

Let $Q_n = (\Gamma_n)$ denote the new connection request. Similar to the SCAS scheme, the eSCAS scheme finds an HS-DSCH $C_{k,n}$ (supporting the transmission rate $R_{k,n}$) such that $R_{k,n} - \sum_{Q_i \in S_{k,n}} \Gamma_i \geq \Gamma_n$, where $S_{k,n}$ is the set containing the IDs of all the connections served by this HS-DSCH. If such an HS-DSCH is not available, then the request is rejected (see 4 in Fig. 3). Otherwise, the assignment, translation, scheduling, and transition procedures are executed to prepare transmission service for this request. The assignment procedure (see 5 in Fig. 3) allocates the selected HS-DSCH to Q_n. The eSCAS scheme looks up the transition table $\Pi_{k,n}$ of $C_{k,n}$ to see whether there exists a previously completed connection Q_i whose Γ_i is the same as that of Q_n (see 6 in Fig. 3). If such Q_i exists, the transition procedure (see 7 in Fig. 3) exercises to reserve TTIs for Q_n by using the same configuration of the previously served connection. Otherwise (i.e., no such Q_i exits), the translation procedure (see 8 in Fig. 3) divides the request Q_n into a set of N subconnection requests $\{q_{n,1}, q_{n,2}, \cdots, q_{n,N}\}$. If Q_n is a two's power connection, the translation procedure is skipped. Then, the scheduling procedure (see 9 in Fig. 3) reschedules the connections (including subconnections). Finally, the Node B starts the transmissions for Q_n. The details of the assignment, translation, scheduling, and transition procedures are given as follows.

A. Assignment Procedure

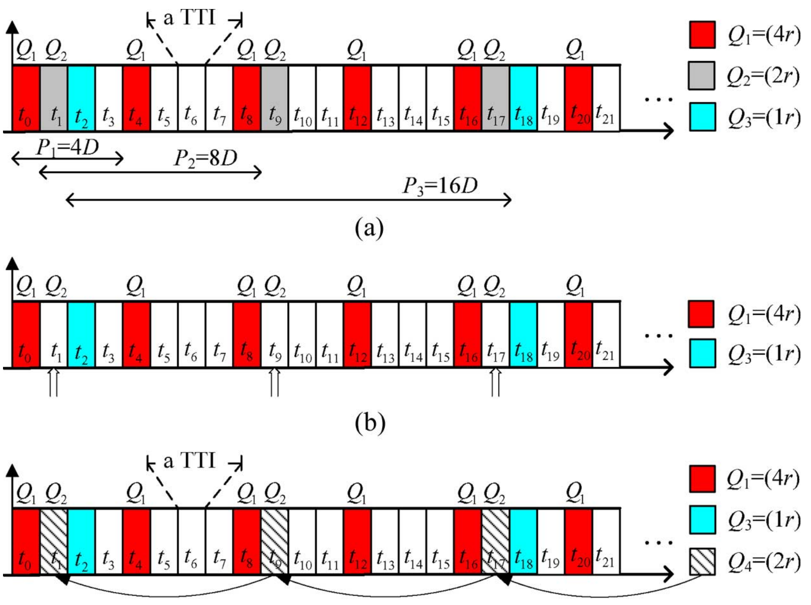
The assignment procedure allocates the selected HS-DSCH $C_{k,n} = (R_{k,n})$ to the new connection $Q_n = (\Gamma_n)$ by adding Q_n into $S_{k,n}$; that is, $S_{k,n} \leftarrow S_{k,n} \cup Q_n$.

B. Translation Procedure

If the Q_n connection is not a two's power connection, this procedure is executed to divide Q_n into a set of N two's power subconnections $\{q_{n,1}, q_{n,2}, \cdots, q_{n,N}\}$, where $\sum_{i=1}^{N} \gamma_{n,i} = \Gamma_n$. To reduce the overhead of scheduling, the translation procedure attempts to find the set of two's subconnections such that N is minimized. Because all subconnections request transmission rates $\gamma_{n,i} = 2^{a_{n,i}}r$ (in bits per second), this procedure adopts the *binary expansion* approach [14], and thus the set with minimized N can be obtained. It first represents Γ_n as the series $\Gamma_n = (a_k 2^k + a_{k-1}2^{k-1} + \cdots + a_1 2^1 + a_0 2^0)r$ (in bits per second), where $a_i \in \{0, 1\}$. For $a_j \neq 0$, the set includes the two's power subconnection with the transmission rate $2^j r$ (in bits per second). Here, we show an example to illustrate the operation of the binary expansion approach. Suppose that $Q_n = (7r)$, which can be represented by $(7)_{10} = (111)_2$. Then, Q_n is divided into the three two's power subconnections: 1) $q_{n,1} = (4r)$; 2) $q_{n,2} = (2r)$; and 3) $q_{n,3} = (1r)$.

C. Scheduling Procedure

After the translation procedure, any arbitrary connection is divided into a set of two's power subconnections. For a two's power connection Q_n, it can be treated as a two's power subconnection $q_{n,1}$, where $\gamma_{n,1} = \Gamma_n$. The scheduling procedure takes the set of two's power subconnections as its


a TTI
Q_1 Q_2 Q_1 Q_1 Q_2 Q_1 Q_1 Q_2 Q_1
t_0 t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} t_{11} t_{12} t_{13} t_{14} t_{15} t_{16} t_{17} t_{18} t_{19} t_{20} t_{21}
. . .
P_1=4D
P_2=8D
P_3=16D
Q_1=(4r)
Q_2=(2r)
Q_3=(1r)
Q_4=(2r)

(a)

(b)

(c)

Fig. 4. (a) Sample schedule of three connections. (b) Connection Q_2 leaves system. (c) New connection Q_4 with the requested transmission rate $2r$ (in bits per second) comes after (b).

inputs. Suppose that Q_n is divided into N subconnections, $q_{n,1}, q_{n,2}, \ldots, q_{n,N}$. Without loss of generality, we assume that $\gamma_{n,1} > \gamma_{n,2} > \ldots > \gamma_{n,N}$. Then, the scheduling procedure reschedules each subconnection $q_{i,j}$ of Q_i for $\gamma_{i,j} < \gamma_{n,1}$. The periodic value calculates as $p_{i,j} = D(R_{k,n}/\gamma_{i,j})$. Then, following the same rescheduling operations in the SCAS scheme (see Section I), the transition table $\Pi_{k,n}$ is reset after the rescheduling operation is performed. Note that the scheduling procedure reschedules only some subconnections currently served, and the rescheduling overhead is expected to be lower.

D. Transition Procedure

This procedure references the information (i.e., Γ_i, O_i, P_i) stored in the transition table to reuse the TTIs for the previously served Q_i to transmit the data of the Q_n connection. Fig. 4 shows an example of the transition procedure. Suppose that initially, there were three two's power connections $Q_1 = (4r)$, $Q_2 = (2r)$, and $Q_3 = (r)$ served by the same HS-DSCH $C_{k,n}$. The TTIs reserved for Q_1, Q_2, and Q_3 are shown in Fig. 4(a). After a while, the transmission for Q_2 was completed. The eSCAS scheme logged the configuration of TTIs reserved for Q_2 into the transition table $\Pi_{k,n}$ of $C_{k,n}$. As shown in Fig. 4(b), the TTIs reserved for Q_2 were set to idle. When a new connection request $Q_4 = (2r)$ is granted, the eSCAS scheme first checks $\Pi_{k,n}$ and finds that the Γ_2 is the same as Γ_4. The eSCAS scheme then reuses the TTIs previously reserved for Q_2 to serve Q_4, as shown in Fig. 4(c).

III. Properties of the eSCAS Scheme

This section explores important properties of the eSCAS scheme. We show that each admitted connection in a channel $C_{k,n}$ will always be assigned the requested transmission rate within any time window of $W = D(R_{k,n}/r)$, where $R_{k,n}/r$ is the number of TTIs available for allocations for a channel $C_{k,n}$ under the consideration of the base rate r (in bits per second), and D is the duration of a TTI. We also show that each channel can be 100% fully utilized. In other words, unless the total transmission rate for connections assigned to a channel $C_{k,n}$ is over its transmission rate $R_{k,n}$, each admitted connection is guaranteed to receive its requested transmission rate within any time window of $W = D(R_{k,n}/r)$.

The transmission rate Γ_i of each connection Q_i could be reencoded in a unique binary representation $\sum_{j=0}^{\lceil \log_2 R_{k,n}/r \rceil} a_{i,j} 2^j r$ (in bits per second), where $a_{i,j}$ is either 0 or 1. When $a_{i,j} = 1$ for a connection Q_i, we say that there is a *subconnection* with the transmission rate $2^j r$ (in bits per second) for Q_i. For example, the transmission rate $\Gamma_i = 9r$ (in bits per second) of a connection Q_i could be reencoded as $2^3 r + r$ (in bits per second), and we say that there are two subconnections with transmission rates $2^3 r$ and r (in bits per second), respectively, for Q_i. In other words, each connection could be considered as a set of subconnections in which each subconnection is with the transmission rate $2^j r$ (in bits per second) for some integer $j \geq 0$. For the rest of this section, the scheduling problem of connections is first rephrased and solved as a scheduling problem of subconnections. We shall show that any schedule of subconnections under the eSCAS scheme would be a legal schedule for their respective connections.

Lemma 1: Given two subconnections $q_i = (\gamma_i)$ and $q_j = (\gamma_j)$ admitted and scheduled by the eSCAS scheme, there will be no overlapping on TTIs scheduled for q_i and q_j.

Proof: Let $p_i = D(R_{k,n}/\gamma_i)$ and $p_j = D(R_{k,n}/\gamma_j)$ denote the periodic values of subconnections q_i and q_j, respectively, and o_i and o_j be the initial times for q_i and q_j, respectively. Suppose that $\gamma_i \geq \gamma_j$ (a similar proof could be done for $\gamma_j \geq \gamma_i$). Because the transmission rate of each subconnection is $2^k r$ (in bits per second) for some integer $k \geq 0$, p_i divides p_j, and let $p_j = h_{i,j} p_i$ for some positive integer $h_{i,j}$. Since the eSCAS scheme always assigns all TTIs located at $o_i + m_i p_i$ for q_i for any integer $m_i \geq 0$, there will be no overlapping on TTIs scheduled for q_i and q_j because $(o_i + m_i p_i)$ could not be equal to $(o_j + m_j p_j) = (o_j + m_j h_{i,j} p_i)$ for any $m_i, m_j \in \mathcal{N}$, where $o_i \neq o_j$. ■

Theorem 1: Given two connections $Q_i = (\Gamma_i)$ and $Q_j = (\Gamma_j)$ admitted and scheduled by the eSCAS scheme, there will be no overlapping on TTIs scheduled for Q_i and Q_j.

Proof: The correctness of this theorem follows directly from Lemma 1 because a connection consists of a collection of subconnections. ■

Lemma 2: Given a collection of subconnections $Q = \{q_1, q_2, \cdots, q_N\}$ of connections for a channel $C_{k,n}$, the eSCAS scheme could always find available TTIs for each subconnection if their total transmission rate is no more than the channel transmission rate $R_{k,n}$, i.e., $R_{k,n} \geq \sum_{q_i \in Q} \gamma_i$.

Proof: It could be proved by contradiction. Let Q be sorted in a nonincreasing order, and the total transmission rate of Q is no more than the channel transmission rate $R_{k,n}$. Let γ_i and $p_i = D(R_{k,n}/\gamma_i)$ denote the transmission rate and the periodic value of each subconnection q_i, respectively. Suppose that $q_j \in Q$ is the first subconnection that could not be scheduled under the eSCAS scheme. Based on the definitions of the eSCAS scheme, the following inequality is true: The number of TTIs allocated to subconnections $q_1, \cdots, q_{j-1}$ within a time frame of size p_i (starting from its initial time o_i) is equal to the following formula:

$$\begin{aligned}\frac{p_j}{p_1} + \frac{p_j}{p_2} + \cdots + \frac{p_j}{p_{j-1}} &= \frac{\gamma_1}{\gamma_j} + \frac{\gamma_2}{\gamma_j} + \cdots + \frac{\gamma_{j-1}}{\gamma_j} \\ &\leq \frac{R_{k,n} - \gamma_j}{\gamma_j} \\ &< \frac{R_{k,n}}{\gamma_j} = \frac{p_j}{D}.\end{aligned}$$

The satisfaction of the aforementioned inequality implies that the eSCAS scheme could always find an available TTI within a time frame of size p_i (starting from its initial time o_i). ■

Theorem 2: Given a collection of N connections $S = \{Q_1 = (\Gamma_1), \cdots, Q_N = (\Gamma_N)\}$, each connection Q_i assigned to the channel $C_{k,n} = (R_{k,n})$ is guaranteed to receive its required transmission rate Γ_i within any time window of size $W = D(R_{k,n}/r)$ if $R_{k,n} \geq \sum_{Q_i \in S} \Gamma_i$.

Proof: Let each connection Q_i consist of a set of c_i subconnections $\{q_{i,j}\}$ such that $\Gamma_i = \sum_{j=1}^{c_i} \gamma_{i,j}$. Since $R_{k,n} \geq \sum_{Q_i \in S} \Gamma_i$, i.e., $R_{k,n} \geq \sum_{Q_i \in S} \sum_{j=1}^{c_i} \gamma_{i,j}$, there is one TTI reserved for each subconnection $q_{i,j}$ within every time frame of size $p_{i,j}$ based on Lemma 1 (nonoverlapping) and 2 (availability). In other words, each subconnection $q_{i,j}$ would have $W/p_{i,j}$ TTIs within any time window of size W. That is, the transmission rate $((W/p_{i,j})/(W/D))R_{k,n} = (D/p_{i,j})R_{k,n} = \gamma_{i,j}$ is guaranteed for each subconnection $q_{i,j}$ within any time window of size W. As a result, each connection Q_i is guaranteed to have the transmission rate $\Gamma_i = \sum_{j=1}^{c_i} \gamma_{i,j}$ within any time window of size W. ■

Corollary 1: Given a collection of N connections $S = \{Q_1 = \{\Gamma_1\}, \cdots, Q_N = \{\Gamma_N\}\}$, each connection Q_i assigned to the channel $C_{k,n} = (R_{k,n})$ is guaranteed to receive its required transmission rate Γ_i within any time window of size $W' = D(R_{k,n}/\min\{\gamma_{i,j}\})$ if $R_{k,n} \geq \sum_{Q_i \in S} \Gamma_i$.

Proof: This corollary could be proved in a way similar to that for Theorem 2 by replacing each occurrence of $D(R_{k,n}/r)$ (i.e., W) with $W' = D(R_{k,n}/\min\{\gamma_{i,j}\})$. ■

IV. Performance Evaluation

As mentioned in [13], the SCAS scheme has lower complexity and higher resource utilization over previous scheduling schemes. With the same characteristic of SCAS, the eSCAS scheme has the same advantages of SCAS. Furthermore, eSCAS overcomes the two's power limitation of SCAS by dividing a connection into several subconnections (see the translation procedure in Section II). However, after the "dividing" operations, there are more connections in the HS-DSCH, which may cause more rescheduling overhead to serve a new connection. With the scheduling and transition procedures, we expect that the rescheduling overhead can be reduced. To clearly study this performance issue, we conduct an analytic model and simulation experiments to investigate the performance for SCAS and eSCAS in terms of the new connection blocking probability and rescheduling overhead defined as follows.

Consider the HS-DSCH supporting the transmission rate $R_{k,n} = 2^{K-k} r$ (in bits per second). Suppose that there are M types of connections $(1 \leq M \leq 2^{K-k})$, and a type i connection $(1 \leq i \leq M)$ requests the transmission rate $i \times r$ (in bits per second). Let $p_{b,i}$ be the blocking probability that a type i connection cannot be served by the HS-DSCH. The average blocking probability P_b is defined as

$$P_b = \frac{\sum_{1 \leq i \leq M} p_{b,i}}{M}. \tag{1}$$

As mentioned in [4], the QoS degradation caused by blocking a high transmission rate connection may be higher than that caused by blocking a low transmission rate connection. Following [4], we define the weighted blocking probability P_{wb} as follows to indicate the overall network QoS:

$$P_{wb} = \frac{\sum_{1 \leq i \leq M} i \times p_{b,i}}{\sum_{1 \leq i \leq} i}. \tag{2}$$

Let N_r denote the average number of the connections (or subconnections) that are rescheduled (i.e., their assigned TTIs are changed) for a newly granted connection. We use N_r to indicate the rescheduling overhead of the SCAS and eSCAS schemes. In the following, we derive P_b and P_{wb} for eSCAS and SCAS. For $1 \leq i \leq M$, suppose that type i connection arrivals form a Poisson process with interarrival rate λ_i, the service times for type i connections have a general distribution with mean $1/\mu_i$, and the total transmission rate supported by the HS-DSCH is $R_{k,n} = 2^{K-k} r$ (in bits per second).

For eSCAS, we consider a stochastic process with state $\mathbf{n} = (m_1, m_2, \ldots, m_M)$, where m_i $(1 \leq i \leq M)$ is the number

TABLE I
VALIDATION OF THE SIMULATION AND THE ANALYTIC MODELS ($R_k = 32r$ b/s; $M = 8$; $\mu_i = \mu$; $\lambda_i = \lambda$)

	eSCAS			**SCAS**		
	Simulation	**Analysis**	**Error**	**Simulation**	**Analysis**	**Error**
$P_b(\lambda = 2\mu/M)$	1.167	1.172	0.42%	3.253	3.234	0.58%
$P_{wb}(\lambda = 2\mu/M)$	1.599	1.593	0.37%	4.127	4.113	0.34%
$P_b(\lambda = 3\mu/M)$	4.039	4.065	0.64%	8.545	8.574	0.33%
$P_{wb}(\lambda = 3\mu/M)$	5.385	5.395	0.18%	10.812	10.821	0.08%
$P_b(\lambda = 4\mu/M)$	8.483	8.481	0.02%	14.900	14.904	0.02%
$P_{wb}(\lambda = 4\mu/M)$	11.053	11.064	0.09%	18.741	18.684	0.30%
$P_b(\lambda = 5\mu/M)$	13.692	13.631	0.45%	21.224	21.131	0.44%
$P_{wb}(\lambda = 5\mu/M)$	17.494	17.545	0.29%	26.433	26.330	0.39%

of type i connections currently served by the HS-DSCH. Then, the state space A for the stochastic process can be expressed as

$$A = \left\{ \mathbf{n} \middle| \sum_{1 \le i \le M} i \times m_i \le 2^{K-k} \right\}.$$

For $1 \le i \le M$, let $\rho_i = \lambda_i / \mu_i$ be the offered load for type i connections. According to Kelly [15] and Zachary [16], the stationary probability of the state $\mathbf{n} = (m_1, m_2, \ldots, m_M)$ can be obtained by

$$p(\mathbf{n}) = G^{-1} \left(\prod_{1 \le i \le M} \frac{\rho_i^{m_i}}{m_i!} \right)$$

where

$$G = \sum_{\mathbf{n} \in A} \left(\prod_{1 \le i \le M} \frac{\rho_i^{m_i}}{m_i!} \right).$$

Let A_i be the set of states where a type i connection request cannot be served by the HS-DSCH, which is resulted from that the left transmission rate $(2^{K-k} - \sum_{1 \le j \le M} j \times m_j) r$ (in bits per second) is less than $i \times r$ (in bits per second). Then, A_i can be expressed as

$$A_i = \left\{ \mathbf{n} \middle| \mathbf{n} \in A \text{ and } 2^{K-k} - \sum_{1 \le j \le M} (j \times m_j) < i \right\}.$$

Then, $p_{b,i}$ can be computed by

$$p_{b,i} = \sum_{\forall \mathbf{n} \in A_i} p(\mathbf{n}). \tag{3}$$

By applying (3) into (1) and (2), we have P_b and P_{wb} for eSCAS as

$$P_b = \frac{\sum_{1 \le i \le M} \sum_{\forall \mathbf{n} \in A_i} p(\mathbf{n})}{M}$$

and

$$P_{wb} = \frac{\sum_{1 \le i \le M} \left(i \times \sum_{\forall \mathbf{n} \in A_i} p(\mathbf{n}) \right)}{\sum_{1 \le i \le M} i}.$$

Consider the stochastic process with state $\mathbf{n} = (m_1, m_2, \ldots, m_M)$ for SCAS. In SCAS, a type i connection arrival consumes the transmission rate $2^{\lceil \log_2 i \rceil} r$ (in bits per second). Then, the state space B of the stochastic process for SCAS can be expressed as

$$B = \left\{ \mathbf{n} \middle| \sum_{1 \le i \le M} 2^{\lceil \log_2 i \rceil} \times m_i \le 2^{K-k} \right\}.$$

Let B_i be the set of states where a type i connection request is rejected due to that the left transmission rate $(2^{K-k} - \sum_{1 \le j \le M} 2^{\lceil \log_2 j \rceil} \times m_j) r$ (in bits per second) is less than $2^{\lceil \log_2 i \rceil} r$ (in bits per second). Then, B_i can be expressed as

$$B_i = \left\{ \mathbf{n} | \mathbf{n} \in B \text{ and } 2^{K-k} - \sum_{1 \le j \le M} \left(2^{\lceil \log_2 j \rceil} \times m_j \right) < 2^{\lceil \log_2 i \rceil} \right\}.$$

Similar to the derivation of P_b and P_{wb} for eSCAS, we have P_b and P_{wb} for SCAS as

$$P_b = \frac{\sum_{1 \le i \le M} \sum_{\forall \mathbf{n} \in B_i} p(\mathbf{n})}{M}$$

and

$$P_{wb} = \frac{\sum_{1 \le i \le M} \left(i \times \sum_{\forall \mathbf{n} \in B_i} p(\mathbf{n}) \right)}{\sum_{1 \le i \le M} i}.$$

Note that in our analysis, P_b and P_{wb} are independent of the service time distribution, and the service time distribution will not affect the performance.

We also conduct simulation experiments in this paper for two purposes: 1) in the simulation, we release the limitation of the analytical model (where we assume that connection interarrival times are exponentially distributed) and 2) our simulation model can be used to study the N_r performance for SCAS and eSCAS. Our simulation model follows the event-driven approach that has been widely adopted in several mobile network studies [13], [17]. Due to the page limitation, the details of the simulation model are not presented in this paper. The results of simulation experiments and the analytical model are validated against each other, as shown in Table I. The table shows that the errors between simulation and analysis results are within 0.7%. The results of simulation and analysis are consistent. The details of the parameter setup in this table will be elaborated later.

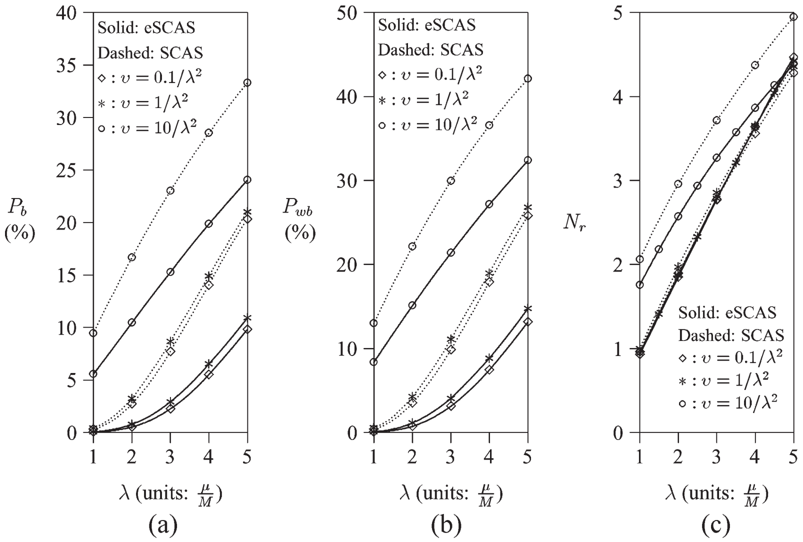


Fig. 5. Performance comparison between SCAS and eSCAS ($R_{k,n} = 128r$ b/s; $M = 32$; $\mu_i = \mu$; $\lambda_i = \lambda$; $v_i = v$).

Based on the simulation experiments, we investigate the P_b, P_{wb}, and N_r performances for eSCAS and SCAS, where the interarrival times for type i connections are set to a Gamma distribution with mean λ_i and variance v_i. The Gamma distribution can be used to approximate many other distributions as well as measured data from real systems [17]–[21]. The total transmission rate $R_{k,n}$ of the HS-DSCH is set to $128r$ (in bits per second). Thirty-two types of connections are considered (i.e., $M = 32$), and a type i $(1 \leq i \leq M)$ connection requests the transmission rate $i \times r$ (in bits per second).

For $1 \leq i \leq M$, we consider the scenario where $\lambda_i = \lambda$, $v_i = v$, and $\mu_i = \mu$. For other scenarios, we observe similar results, which are not presented in this paper. To simplify our discussion, λ is normalized by the service time μ. In the simulation experiments, λ ranges from $1\mu/M$ to $5\mu/M$. The variance v is set to $v = 0.1/\lambda^2$, $v = 1/\lambda^2$, and $v = 10/\lambda^2$.

A. P_b and P_{wb} Performance

Fig. 5(a) and (b) compares the blocking probability P_b and the weighted blocking probability P_{wb} for SCAS and eSCAS. These two figures show that P_b and P_{wb} for SCAS and eSCAS have the same trend, and eSCAS outperforms SCAS significantly. As v increases, both P_b and P_{wb} increase. When $v = 1/\lambda^2$ or $v = 0.1/\lambda^2$, the P_b and P_{wb} values for SCAS are almost twice of that for eSCAS, for example, when $v = 1/\lambda^2$ and $\lambda = 5\mu/M$, P_b for eSCAS is 10.9%, and P_b for SCAS is 21.0%. When $v = 10/\lambda^2$, the difference between eSCAS and SCAS becomes smaller. For example, when $v = 10/\lambda^2$ and $\lambda = 5\mu/M$, P_b for eSCAS is 24.0%, P_b for SCAS is 33.2%, and P_b for eSCAS is 72% of that for SCAS. When the variance v is larger than $1/\lambda^2$, more short connection interarrival times are observed, and more connections arrive in a short period. These connection arrivals compete for the transmission rate, and higher blocking probabilities are observed. On the other hand, when $v \leq 1/\lambda^2$, the connection interarrival times are clustered to the mean of the distribution, and the connections arrive stably. The competition is released, and lower blocking probabilities are expected. Since SCAS must reserve transmission rate [with a two's power number of r (in bits per second)] for a non-two's power connection, some transmission rate is wasted. On the other hand, eSCAS releases this limitation. Hence, we observe that eSCAS has better P_b and P_{wb} performance than that SCAS does, and the improvement of P_b and P_{wb} for eSCAS over that for SCAS is significant.

B. N_r Performance

Fig. 5(c) investigates the rescheduling overhead N_r for eSCAS and SCAS. The figure shows that for both two schemes, N_r increases as λ increases. When λ increases, it is likely that more connections are served when a new connection is accommodated, and more connections are affected by the rescheduling operation. We also observe that for both schemes, the N_r values when $v = 10/\lambda^2$ are larger than that when $v = 1/\lambda^2$ or $v = 0.1/\lambda^2$. As mentioned previously, when $v > 1/\lambda^2$, more short connection interarrival times are observed, and it is likely that more connections are served when a new connection gets service. Thus, the rescheduling operation may affect more served connections. Another important phenomenon observed is that the N_r performances for eSCAS and SCAS are almost identical when $v \leq 1/\lambda^2$. When $v = 10/\lambda^2$, eSCAS outperforms SCAS in terms of the N_r performance. This is due to that when $v > 1/\lambda^2$ (i.e., larger variance), more connections arrive in a short period. It is more likely that the configuration information of the previously served connections can be referenced, and the new connection can be served by using this configuration. The rescheduling overhead drops; hence, we observe better N_r performance for eSCAS than that for SCAS when v is large.

To summarize, the proposed eSCAS scheme can significantly lower the blocking probabilities (i.e., P_b and P_{wb}) without increasing the rescheduling overhead.

V. CONCLUDING REMARK

This paper proposed a time-division-based cyclic scheduling scheme eSCAS to allocate a UMTS HS-DSCH to serve different connections requesting different transmission rates. The eSCAS scheme is enhanced from the SCAS scheme proposed in our previous study by releasing the two limitations of SCAS, including two's power limitation and rescheduling overhead. We examined the properties of the eSCAS scheme, and then formally proved the correctness of eSCAS. An analytical model and simulation experiments were conducted to compare the performance of eSCAS and SCAS in terms of the blocking probabilities P_b and P_{wb} and rescheduling overhead N_r. Our study shows that the proposed eSCAS scheme can significantly decrease the blocking probabilities P_b and P_{wb} without increasing rescheduling overhead N_r, and when the variance of the connection interarrival times is large, eSCAS gains better N_r performance that SCAS does (i.e., lower rescheduling overhead).

ACKNOWLEDGMENT

The authors would like to thank the editor and the two anonymous reviewers for their valuable comments.

REFERENCES

[1] 3GPP, *General Packet Radio Service (GPRS); Service Description; Stage 2*, 2001. version 4.1.0 (2001-06).

[2] 3GPP, *3rd Generation Partnership Project; Radio Interface Protocol Architecture*, 2002. version 4.3.0 (2002-06).

[3] H. Holma and A. Toskala, *WCDMA for UMTS*, 2nd ed. Hoboken, NJ: Wiley, 2002.

[4] P. Lin, C.-H. Gan, and C.-C. Hsu, "OVSF code channel assignment with dynamic code set and buffering adjustment for UMTS," *IEEE Trans. Veh. Technol.*, vol. 54, no. 2, pp. 591–602, Mar. 2005.

[5] 3GPP, *High Speed Downlink Packet Access (HSDPA); Overall Description; Stage 2 (Release 5)*, 2004. version 5.7.0 (2004-12).

[6] 3GPP, *Technical Specification Group Radio Access Network; High Speed Downlink Packet Access: Iub/Iur Protocol Aspects (Release 5)*, 2002. version 5.1.0 (2002-06).

[7] H. Fattah and C. Leung, "An overview of scheduling algorithms in wireless multimedia networks," *IEEE Wireless Commun.*, vol. 9, no. 5, pp. 76–83, Oct. 2002.

[8] A.-K. Parekh and R.-G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: The single-node case," *IEEE/ACM Trans. Netw.*, vol. 1, no. 3, pp. 344–357, Jun. 1993.

[9] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," *J. Internetworking Res. Experience*, vol. 1, no. 1, pp. 3–26, Oct. 1990.

[10] L. Zhang, "Virtual clock: A new traffic control algorithm for packet switching networks," in *Proc. ACM SIGCOMM*, 1990, pp. 19–29.

[11] A. Kuurne and A.-P. Miettinen, "Weighted round robin scheduling strategies in (E)GPRS radio interface," in *Proc. IEEE Veh. Technol. Conf.*, Sep. 2004, vol. 5, pp. 3155–3159.

[12] P. Lin, C.-H. Gan, N.-C. Perng, T.-W. Kuo, and C.-C. Hsu, "Time division based shared channel allocation algorithm for UMTS," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Mar. 2004, vol. 2, pp. 717–721.

[13] C.-H. Gan, P. Lin, N.-C. Perng, T.-W. Kuo, and C.-C. Hsu, "Scheduling for time-division based shared channel allocation for UMTS," *Springer/ACM Wireless Netw.*, vol. 13, no. 2, pp. 189–202, Apr. 2007.

[14] K.-H. Rosen, *Discrete Mathematics and Its Applications*, 3rd ed. New York: McGraw-Hill, 1995.

[15] F.-P. Kelly, "Loss networks," *Ann. Appl. Probab.*, vol. 1, no. 3, pp. 319–378, 1991.

[16] S. Zachary, "On blocking in loss networks," *Adv. Appl. Probab.*, vol. 23, no. 2, pp. 355–372, Jun. 1991.

[17] P. Lin, Y.-B. Lin, C.-H. Gan, and J.-Y. Jeng, "Credit allocation for UMTS prepaid service," *IEEE Trans. Veh. Technol.*, vol. 55, no. 1, pp. 306–316, Nov. 2005.

[18] P. Lin and Y.-B. Lin, "Channel allocation for GPRS," *IEEE Trans. Veh. Technol.*, vol. 50, no. 2, pp. 375–387, Mar. 2001.

[19] P. Lin, "Channel allocation for GPRS with buffering mechanisms," *Springer/ACM Wireless Netw.*, vol. 9, no. 5, pp. 431–441, Sep. 2003.

[20] P. Lin, Y.-B. Lin, and I. Chlamtac, "Modeling frame synchronization for UMTS high-speed downlink packet access," *IEEE Trans. Veh. Technol.*, vol. 50, no. 1, pp. 132–141, Jan. 2003.

[21] P. Lin and G.-H. Tu, "An improved GGSN failure restoration mechanism for UMTS," *Springer/ACM Wireless Netw.*, vol. 12, no. 1, pp. 91–103, Feb. 2006.

Chai-Hien Gan (M'05) was born in Malaysia in 1971. He received the B.S. degree in computer science from Tamkang University, Tamsui, Taiwan, R.O.C., in 1994 and the M.S. and Ph.D. degrees in computer science and information engineering from the National Taiwan University, Taipei, Taiwan, in 1996 and 2005, respectively.

Since March 2005, he has been a Research Assistant Professor in the Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan. His current research interests include wireless mesh networks, mobile computing, personal communications services, and wireless Internet.

Nei-Chiung Perng received the B.S. and M.S. degrees in computer and information science from the National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 1999 and 2001, respectively, and the Ph.D. degree in computer science and information engineering from the National Taiwan University, Taipei, Taiwan, in 2006.

He is currently with Genesyslogic, Inc., Taipei. His research interests include real-time systems and scheduling algorithms.

Phone Lin (M'02–SM'06) received the B.S. degree in computer science and information engineering and the Ph.D. degree from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 1996 and 2001, respectively.

From August 2001 to July 2004, he was an Assistant Professor in the Department of Computer Science and Information Engineering (CSIE), National Taiwan University (NTU), Taipei, Taiwan. Since August 2004, he has been an Associate Professor in the Department of CSIE and in the Graduate Institute of Networking and Multimedia, NTU. He is a Guest Editor of the ACM/Springer MONET Special Issue on Wireless Broad Access. He is also an Associate Editorial Member of the *WCMC Journal*. His current research interests include personal communications services, wireless Internet, and performance modeling.

Dr. Lin is an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and a Guest Editor of the IEEE Wireless Communications Special Issue on Mobility and Resource Management. He received the Research Award for Young Researchers from the Pan Wen-Yuan Foundation in Taiwan in 2004, the K. T. Li Young Researcher Award honored by ACM Taipei Chapter in 2004, the Wu Ta You Memorial Award of the National Science Council (NSC) in Taiwan in 2005, and the Fu Suu-Nien Award of NTU in 2005 for his research achievements. He is listed in *Who's Who in Science and Engineering* in 2006.

Tei-Wei Kuo (SM'02) received the B.S.E. degree in computer science and information engineering from the National Taiwan University, Taipei, Taiwan, R.O.C., in 1986 and the M.S. and Ph.D. degrees in computer science from the University of Texas, Austin, in 1990 and 1994, respectively.

He is currently a Professor in and the Chairman of the Department of Computer Science and Information Engineering, National Taiwan University. Since February 2006, he has been a Deputy Dean of the College of Electrical Engineering and Computer Science, National Taiwan University. He is the author of the book *Real-Time Database Systems: Architecture and Techniques* (Kluwer Academic Publishers) and over 120 technical papers published or accepted for publication in international journals and conference proceedings. His research interests include embedded systems, real-time operating systems, and real-time database systems. He has served as an Associate Editor of the *Journal of Real-Time Systems* (SCI) since 1998 and has been in the editorial board of the *Journal of Information Science and Engineering* (EI) since 2005 and the *Journal of Embedded Computing* since 2004.

Dr. Kuo is a Program Committee Member of many international conferences around the world. He is a Program Cochair of the Seventh International Conference on Real-Time Computing Systems and Applications (RTCSA 2000), December 12–14, 2000, Cheju Island, Korea, a Program Cochair of the IEEE Real-Time Technology and Applications Symposium, Taipei, Taiwan, in 2001, and the Program Chair for Asia and the Far East of the Ninth International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS 2001) in San Francisco, CA, in 2001. Since 2003, he has been a member of the Steering Committee of IEEE RTCSA. He is now an Executive Committee Member of the IEEE Technical Committee on Real-Time Systems and the Steering Committee Chair of IEEE RTCSA. He received several research awards in Taiwan, including the Distinguished Research Award from the R.O.C. National Science Council in 2003 and the Young Scholar Research Award from Academia Sinica, Taiwan, in 2001. He received the R.O.C. Ten Outstanding Young Persons Award in 2004 for his research achievements and contributions in Taiwan.