

Modeling Frame Synchronization for UMTS High-Speed Downlink Packet Access

Phone Lin, *Member, IEEE*, Yi-Bing Lin, *Fellow, IEEE*, and Imrich Chlamtac, *Fellow, IEEE*

Abstract—Third Generation Partnership Project (3GPP) specification TR 25.950 proposed *High-Speed Downlink Packet Access for the Universal Mobile Telecommunication System (UMTS)*. In this mechanism, an active set of cells is defined for every *User Equipment (UE)* communication session. The cell with the best wireless link quality (called the serving cell) in the active set is selected for communication with the UE. When the wireless link quality of the old serving cell degrades below some threshold, a new serving cell in the active set is selected to continue the communication session. Our previous work proposed a High-Speed Downlink Packet Access (HSDPA) overflow control scheme with four frame synchronization algorithms to switch the serving cell, and formally proved the correctness of the scheme. In this paper, we propose an analytic model to investigate the performance of these frame synchronization algorithms, and show how the user movement patterns affect the control message delivery costs of these algorithms.

Index Terms—Frame synchronization, High-Speed Downlink Packet Access (HSDPA), Universal Mobile Telecommunication System (UMTS).

I. INTRODUCTION

UNIVERSAL Mobile Telecommunication System (UMTS) [6] supports high data transmission rate for mobile users. As shown in Fig. 1, the UMTS system consists of the *Terrestrial Radio Access Network (UTRAN)* and the core network. The UTRAN consists of radio base stations (Node Bs) and *Radio Network Controllers (RNCs)*. The RNC and Node Bs are connected through an asynchronous transfer mode (ATM) network. The radio coverage area of a transmitter in a Node B is called a *cell*. In this paper, we will use the term “cell” to represent “Node B.” A *User Equipment (UE)* communicates with a cell through the air interface Uu [1]. In a UE communication session, several cells (Node Bs) are defined as an *active set*. If the quality of the wireless link between the UE and a cell is above some threshold, then this cell is included in the active set. When

the quality of the wireless link of a cell in the active set is below some threshold, then the cell is removed from the active set. The size of an active set is typically limited to three cells. In a standard UTRAN [5], multiple radio transmission paths exist between the UE and all cells in the active set. This mechanism does not support high-speed downlink transmission because multiple radio links to an UE may increase the overall interference within the UTRAN, and thus the data transmission rate decreases.

Third Generation Partnership Project (3GPP) specification TR 25.950 [4] proposed a mechanism to support *High-Speed Downlink Packet Access (HSDPA)* [4], [2], [3], where a UE only communicates with one cell (called the serving cell) in the active set. This serving cell is selected by the *Fast Cell Selection (FCS)* criteria described in 3GPP TR 25.848 [3]. Other cells in the active set will not transmit packet frames to the UE. Therefore, the interference within a cell is potentially decreased, and the transmission rate for a communication session is increased. Several feasibility studies have been contributed to HSDPA [16], [19], [9], [12], [14], [13]. The topics addressed in these studies include *Adaptive Modulation and Coding*, *Hybrid Automatic Repeat Request*, and *Packet Scheduler*. In this paper, we will focus on the frame synchronization issue (to be elaborated later) for FCS.

In HSDPA, the RNC sends the packet frames to all cells in the active set. For the serving cell, the packet frames are forwarded to the UE. For each nonserving cell, the packet frames are queued in a buffer. If the link quality for high-speed downlink transmission from the serving cell to the UE degrades below some threshold, the UE selects the best cell in the active set as the new serving cell. Then the next packet frames are transmitted from the new serving cell to the UE. When the UE switches to a new serving cell for downlink packet access, the new serving cell should be informed of the status of the buffer (i.e., the number of packet frames received by the UE) in the old serving cell, and adjust its buffer appropriately, so that the next packet frame is correctly delivered from the new serving cell to the UE. This action is referred to as *frame synchronization* [4]. In HSDPA, the nonserving cells do not send packet frames to the UE, and their buffers may overflow during the downlink transmission. A mechanism is required to avoid buffer overflow at these nonserving cells.

The buffer overflow issue was not addressed in 3GPP TR 25.950. In [17], we proposed an overflow control scheme with four frame synchronization algorithms and proved the correctness of these algorithms. This paper proposes analytic and simulation models to investigate the performance of these frame synchronization algorithms. The abbreviations and notations used in this paper are listed in the Appendix.

Manuscript received May 21, 2002; revised September 6, 2002. The work of P. Lin was supported in part by the NSC, R.O.C., under Contract NSC91-213-E-002-065, FarEastone, and CCL/ITRI under Contract 2H-4. The work of Y.-B. Lin was sponsored in part by MOE Program for Promoting Academic Excellence of Universities under the grant number 89-E-FA04-1-4, IIS, Academia Sinica, Intel, FarEastone, CCL/ITRI, the Lee and MTI Center for Networking Research, NCTU.

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

Y.-B. Lin is with the Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu, Taiwan, R.O.C. (e-mail: liny@csie.nctu.edu.tw).

I. Chlamtac is with the Center for Advanced Telecommunications Systems and Services, The University of Texas at Dallas, Richardson, TX 75083-0688 USA (e-mail: chlamtac@utdallas.edu).

Digital Object Identifier 10.1109/TVT.2002.807220

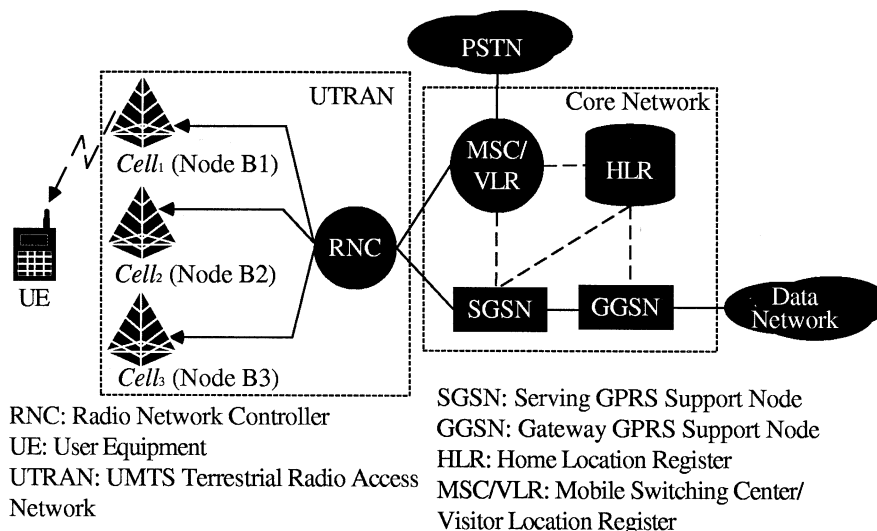


Fig. 1. The UMTS architecture.

II. THE OVERFLOW CONTROL SCHEME FOR HSDPA

This section describes the overflow control scheme and the four frame synchronization algorithms proposed in [17]. We first describe the overflow control scheme (consisting of algorithms OFC1–OFC4) exercised during downlink transmission. Then, the four frame synchronization algorithms proposed in [17] are presented.

A. The Overflow Control (OFC) Algorithms

To exercise HSDPA, every cell $Cell_i$ in the active set maintains a first in first out (FIFO) buffer. The UE maintains a counter CR_{UE} to indicate the number of received packet frames. When the UE switches the wireless link from the old serving cell to the new serving cell, the CR_{UE} value is sent to the new serving cell for frame synchronization. $Cell_i$ maintains the CS_i counter to count the number of packet frames that have been processed by $Cell_i$. If $Cell_i$ is a serving cell, then CS_i is the number of packet frames that have been received by the UE. If $Cell_i$ is a nonserving cell, then CS_i is the number of packet frames deleted from the buffer. To initiate HSDPA, the UE selects the serving cell based on the fast cell selection criteria as described in the previous section, and CR_{UE} and CS_i values are initially set to zero. The relationship among the overflow control algorithms is illustrated in Fig. 2. These algorithms are executed by the RNC (i.e., OFC1), the serving cell (i.e., OFC2 and OFC3), and the nonserving cell (i.e., OFC4). The OFC1–OFC4 algorithms are described as follows.

Algorithm OFC1 is exercised at the RNC [see Fig. 2(1)]. A window-based flow control algorithm with window size w is used for downlink transmission from the RNC to the cells in the active set. The ATM AAL2 is adopted for links between the cells and the RNC [5], which is considered reliable. The RNC sends a packet frame to every cell in the active set. After sending all packet frames of the current window, the RNC must wait for an ACK message from the serving cell $Cell_s$ (see Algorithm OFC2) before it can proceed to send packet frames in the next window.

Algorithm OFC2 is exercised at $Cell_s$ [see Fig. 2(2)] to perform flow control between the RNC and $Cell_s$. When $Cell_s$ has received all packet frames of the current window, it checks the free space of the buffer. If there is enough free space to accommodate the packet frames in the next window, $Cell_s$ replies an ACK message to the RNC. Otherwise, $Cell_s$ suspends the ACK message sending until the buffer has enough free space.

Algorithm OFC3 performs flow control between $Cell_s$ [see Fig. 2(3)] and the UE [see Fig. 2(4)], where the *stop-and-wait hybrid automatic-repeat request (ARQ)* (SAW-Hybrid ARQ) [20] is used for flow control. If the UE receives an error packet frame from $Cell_s$, it replies a NACK message to $Cell_s$ through the uplink *dedicated physical control channel (DPCCH)* in the air interface, and $Cell_s$ retransmits the packet frame to the UE. If a correct packet frame is received, the UE increments CR_{UE} by one and replies $Cell_s$ an ACK message through the uplink DPCCH. Then, $Cell_s$ deletes the last transmitted packet frame from the buffer, and increments CS_s by one. This algorithm guarantees that the CR_{UE} value is the same as the CS_s value in $Cell_s$.

Algorithm OFC4 is exercised by a nonserving cell $Cell_i$ [see Fig. 2(5)] to process the packet frames received from the RNC. When $Cell_i$ receives a packet frame, it checks if the buffer overflows. If so, it deletes a packet frame in the front of the buffer, adds the received packet frame at the tail of the buffer, and then increments CS_i by one. Otherwise (i.e., the buffer does not overflow), it adds the received packet frame at the tail of the buffer.

In [17], we proved that for a nonserving cell $Cell_i$, $CS_i \leq CR_{UE}$. We also proved that with an appropriate frame synchronization algorithm (to be presented in the next subsection), OFC4 can safely delete packet frames in the buffer of a nonserving cell, and after the nonserving cell becomes the serving cell, no packet frame is lost.

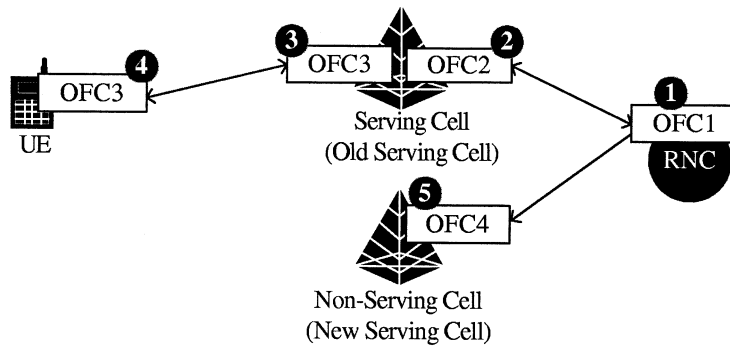


Fig. 2. The relationship among the OFC1–OFC4 algorithms.

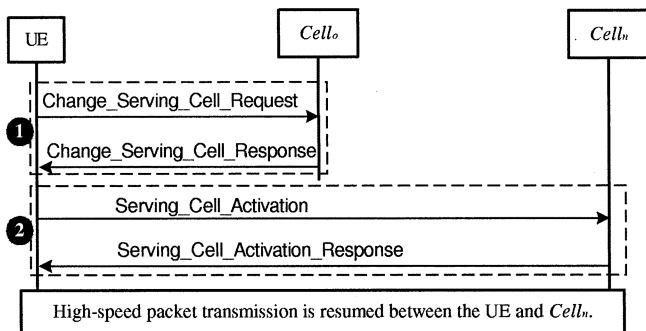


Fig. 3. The control message flow for the BFS algorithm.

B. The Frame Synchronization Algorithms

If the wireless link quality between the serving cell and the UE degrades below some threshold, the UE selects another cell in the active set as the new serving cell. Let the old serving cell and the new serving cell be $Cell_o$ and $Cell_n$, respectively. In [17], we proposed four alternatives for frame synchronization.

Basic Frame Synchronization (BFS): The frame synchronization information (i.e., CR_{UE}) is carried through the uplink DPCCH's in the air interface. As shown in Fig. 3, the algorithm consists of two steps.

Step 1. The UE and $Cell_o$ first exchange the `Change_Serving_Cell_Request` and `Change_Serving_Cell_Response` message pair. At this point, $Cell_o$ stops downlink transmission to the UE and then executes OFC4 to process the packet frames received from the RNC.

Step 2. The UE and $Cell_n$ exchange the `Serving_Cell_Activation` (containing the CR_{UE} value) and `Serving_Cell_Activation_Response` message pair, which indicates that $Cell_n$ will be the serving cell. In this step, $Cell_n$ deletes $CR_{UE} - CS_n$ packet frames in the front of the buffer, and then executes OFC2 to handle the next packet frames received from the RNC.

In [17], we have proved that after $Cell_n$ has deleted $CR_{UE} - CS_n$ packet frames, the number of packet frames received by the UE is the same as that processed by the new serving cell (i.e., no packet frame is lost during the switching of the wireless link). In Step 2 of BFS, the `Serving_Cell_Activation` message is carried through uplink DPCCH in the air interface. The information that

can be carried by one uplink DPCCH is limited. When the CR_{UE} value exceeds the size that can be transmitted through a uplink DPCCH, more than one uplink DPCCH transmissions are required to deliver the CR_{UE} value.

Network Frame Synchronization (NFS): The frame synchronization information (i.e., CS_o) is delivered through the ATM network. Fig. 4 illustrates the message flow for NFS, where three steps are executed.

Step 1. The UE and $Cell_o$ exchange the `Change_Serving_Cell_Request` and `Change_Serving_Cell_Response` message pair. Then $Cell_o$ becomes a non-serving cell as described in Step 1 of BFS.

Step 2. When $Cell_o$ receives the `Change_Serving_Cell_Request` message, it sends the frame synchronization information CS_o to $Cell_n$ by exchanging the `Change_Serving_Cell_Request` and `Change_Serving_Cell_Response` message pair through the RNC.

Step 3. The UE and $Cell_n$ exchange the `Serving_Cell_Activation` and `Serving_Cell_Activation_Response` message pair. At this point, $Cell_n$ deletes $CS_o - CS_n$ packet frames in the front of the buffer, and then becomes the serving cell. Note that the message formats of this step are the same as that in Step 2 of BFS except that the CR_{UE} field is void in this step.

Combined BFS and NFS (CFS): From the descriptions for BFS and NFS, it is clear that if the CR_{UE} value can be fitted in one DPCCH, then the number of control messages exchanged in BFS is less than that in NFS. On the other hand, if multiple DPCCHs are required to deliver CR_{UE} in BFS, then the control message delivery cost of BFS is higher than that of NFS (because the DPCCH delay is anticipated longer and less reliable than that of message delivery in the wireline ATM network). CFS combines BFS with NFS to take advantages of both algorithms. In this algorithm, $Cell_o$ determines whether BFS or NFS should be exercised when it receives the `Change_Serving_Cell_Request` message. Let N^* be the maximum number that can be carried through one uplink DPCCH transmission. If CS_o (which is the same as CR_{UE}) is no larger than N^* , then $Cell_o$ triggers to execute BFS, and CR_{UE} is sent to $Cell_n$ through messages in Fig. 3(2). Otherwise (i.e., $CS_o > N^*$), $Cell_o$ executes

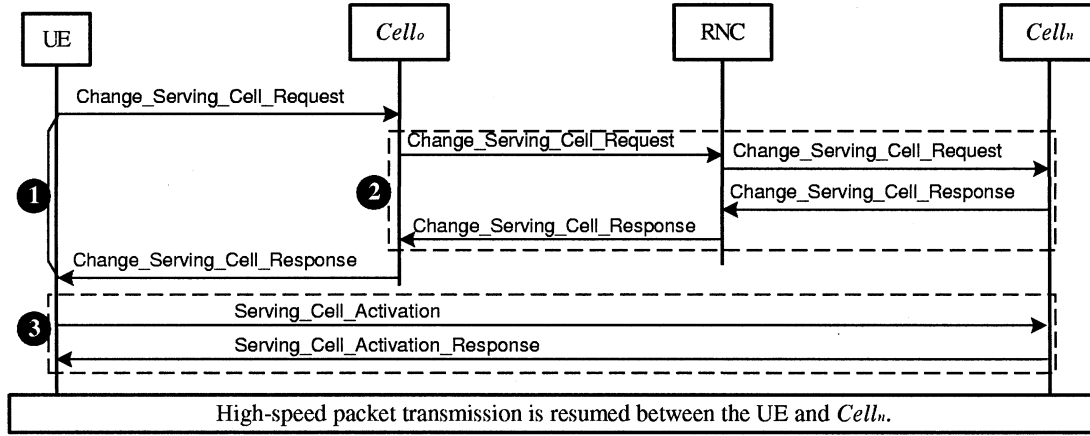


Fig. 4. The control message flow for the NFS algorithm.

NFS by sending the CS_o value to $Cell_n$ through messages in Fig. 4(2).

CFS With Counter Reset (CFSCR): Consider an HSDPA transmission session. Suppose that J frame synchronizations are performed in this session (i.e., the UE switches serving cells J times). At the j th-frame synchronization ($1 \leq j \leq J$), if more than N^* packet frames have been transmitted, then four more control message exchanges are required [see (2) in Fig. 4] when the CFS algorithm is exercised. To reduce the possibility of performing these extra message exchanges, we can reset counters CR_{UE} and CS_i by subtracting a number θ after the UE switches to a new serving cell, where

$$\theta = \left\lfloor \frac{CS_o}{w} \right\rfloor w. \quad (1)$$

In [17], we proved that the θ value computed in (1) guarantees that after counter reset, the overflow control between the RNC and the serving cell operates correctly.

III. ANALYTIC ANALYSIS

This section investigates how traffic patterns (the session time and the packet frame arrival rate) and the serving cell residence time (the period in which the UE resides in a serving cell and receives downlink packet frames from that cell) affect the performance of BFS, NFS, CFS, and CFSCR. Specifically, when the UE switches from the j th cell to the $j + 1$ th cell during a communication session, we compute the number $n_{r,j}$ of control messages delivered in the air interface (between the UE and the cells) and the number $n_{n,j}$ of control messages delivered in the ATM network (among the new serving cell, the old serving cell, and the RNC). Let d_r and d_n be the expected transmission delays in the air interface and the ATM network, respectively. Then the net cost $d_{s,j}$ for frame synchronization and cell switching to the $j + 1$ th cell can be expressed as

$$d_{s,j} = n_{r,j} d_r + n_{n,j} d_n. \quad (2)$$

Let packet frame arrivals to the UE in a downlink transmission session be a Poisson stream with rate λ . Let the serving cell

residence time t of a UE be a random variable with the density function $f(t)$, Laplace transform $f^*(s)$, and the expected value $1/\mu$. If the downlink transmission session is not completed after the UE switches to the $j + 1$ th serving cell, then the session age t_j (the interval between the start of downlink transmission and when the transmission is switched to the $j + 1$ th serving cell) is a random variable with the density function $r_j(t)$ and the Laplace transform

$$r_j^*(s) = [f^*(s)]^j. \quad (3)$$

Let $\Pr[K = k, J = j]$ be the probability that the transmission session is not complete at the j th serving cell, and during t_j , there are k packet frames received by the UE. Then we have

$$\begin{aligned} \Pr[K = k, J = j] &= \int_{t=0}^{\infty} \left[\frac{(\lambda t)^k}{k!} \right] e^{-\lambda t} r_j(t) dt \\ &= \left(\frac{\lambda^k}{k!} \right) \left[(-1)^k \frac{d^k r_j^*(s)}{ds^k} \right] \Bigg|_{s=\lambda}. \end{aligned} \quad (4)$$

Based on (4) and (2), we derive the $d_{s,j}$ cost for BFS, NFS, and CFS. The $d_{s,j}$ cost for CFSCR can also be evaluated by using (4) through extra tedious derivation. For the demonstration purpose, we only present the CFSCR performance through the simulation experiments.

Algorithm BFS: Suppose that in an uplink DPCCCH transmission, b bits are used to keep the frame synchronization information; that is

$$N^* = 2^b - 1 \quad \text{or} \quad b = \log_2(N^* + 1). \quad (5)$$

The number B_{UE} of bits required to carry the CR_{UE} value can be expressed as

$$B_{UE} = \begin{cases} \lceil \log_2(CR_{UE} + 1) \rceil, & \text{if } CR_{UE} \geq 1 \\ 1, & \text{if } CR_{UE} = 0. \end{cases} \quad (6)$$

When the UE switches from the j th cell to the $j+1$ th cell, BFS requires M uplink DPCCCH transmissions, where M is expressed as

$$M = \left\lceil \frac{B_{UE}}{b} \right\rceil. \quad (7)$$

For $j \geq 0$, let $\Pr[M = m, J = j]$ be the probability that there are m uplink DPCCCH transmissions when the UE switches to the $j+1$ th cell. From (4)–(7), we have

$$\Pr[M = m, J = j] = \begin{cases} \sum_{k=0}^{2^b-1} \Pr[K = k, J = j], & \text{if } m = 1 \\ \sum_{k=2^{(m-1)b}}^{2^{mb}-1} \Pr[K = k, J = j], & \text{if } m > 1 \end{cases}$$

and the average number of DPCCCH transmissions in BFS is

$$E[M, J = j] = \sum_{m=1}^{\infty} m \Pr[M = m, J = j]. \quad (8)$$

In the BFS message flow (see Fig. 3), it is clear that $n_{r,j} = 0$ and

$$d_{s,j} = (3 + E[M, J = j]) d_r = \left(3 + \sum_{m=1}^{\infty} m \Pr[M = m, J = j] \right) d_r. \quad (9)$$

Algorithm NFS: When the UE switches from the j th to the $j+1$ th cell, we have $n_{r,j} = 4$ and $n_{n,j} = 4$ as illustrated in Fig. 4. Therefore

$$d_{s,j} = 4d_r + 4d_n. \quad (10)$$

Algorithm CFS: If $CR_{UE} \leq N^*$, then BFS is executed, and four messages are exchanged among the UE, Cell_o , and Cell_n through the air interface. Otherwise (i.e., $CR_{UE} > N^*$), NFS is exercised. From (9) and (10), $d_{s,j}$ is expressed as

$$\begin{aligned} d_{s,j} &= 4d_r \Pr[CR_{UE} \leq N^*, J = j] + (4d_r + 4d_n) \\ &\quad \times (1 - \Pr[CR_{UE} \leq N^*, J = j]) \\ &= 4d_r + 4d_n(1 - \Pr[CR_{UE} \leq N^*, J = j]) \end{aligned} \quad (11)$$

where

$$\Pr[CR_{UE} \leq N^*, J = j] = \sum_{k=0}^{N^*} \Pr[K = k, J = j].$$

To compute (4), we assume that $f(t)$ is a Gamma density function with mean $1/\mu$, standard deviation σ , and the Laplace transform

$$f^*(s) = \left(\frac{\mu}{\mu + \mu^2 \sigma^2 s} \right)^{(1/(\mu\sigma)^2)}. \quad (12)$$

TABLE I
THE $d_{s,j}$ VALUES FOR CFS: SIMULATION VERSUS ANALYSIS
($d_r = 5d_n$; $N^* = 1023$; $j = 3$)

λ/μ	Simulation	Analysis	Error
500	22.6597 d_n	22.6548 d_n	2.162×10^{-4}
1000	23.6607 d_n	23.6606 d_n	4.226×10^{-6}
1500	23.8733 d_n	23.8715 d_n	7.539×10^{-5}
2000	23.9391 d_n	23.9386 d_n	2.088×10^{-5}
2500	23.9596 d_n	23.966 d_n	2.712×10^{-4}

Note that our derivation applies to any cell residence time distributions whose Laplace transforms have close forms. The Gamma distribution is selected to represent the cell residence time distribution because this distribution can be used to approximate many distributions as well as the measured data obtained from PCS fields [15], [8], [10], [11]. By applying (12) into (3), we have

$$r_j^*(s) = \left(\frac{\mu}{\mu + \mu^2 \sigma^2 s} \right)^{(j/(\mu\sigma)^2)}. \quad (13)$$

By substituting (13) into (4), $\Pr[K = k, J = j]$ is rewritten as

$$\begin{aligned} \Pr[K = k, J = j] &= \left(\frac{\lambda^k}{k!} \right) \left[(-1)^k \frac{d^k}{ds^k} \left(\frac{\mu}{\mu + \mu^2 \sigma^2 s} \right)^{(j/(\mu\sigma)^2)} \right] \Big|_{s=\lambda} \\ &= \left(\frac{j}{(\mu\sigma)^2} + k - 1 \right) \binom{j}{k} \left(\frac{\mu}{\mu + \mu^2 \sigma^2 \lambda} \right)^{(j/(\mu\sigma)^2)} \\ &\quad \times \left(\frac{\mu^2 \sigma^2 \lambda}{\mu + \mu^2 \sigma^2 \lambda} \right)^k. \end{aligned} \quad (14)$$

From (9), (11), and (14), $d_{s,j}$ for BFS and CFS can be computed. We also conduct simulation experiments to validate the analytic results. The simulation technique used in this paper is similar to the one described in [18], and the details are omitted. Table I lists the $d_{s,j}$ values of CFS obtained from simulation and analysis. In this table, the cell residence times are exponentially distributed. The table indicates that the analytic results match closely with the simulation data. For gamma cell residence time distributions, similar results are observed and are not presented here.

IV. NUMERICAL EXAMPLES

In this section, we use numerical examples to investigate the performance of the four frame synchronization algorithms.

Effects of λ/μ and j on Algorithms BFS, NFS, and CFS: Fig. 5 plots $d_{s,j}$ as functions of λ/μ and j for BFS, NFS, and CFS, where $d_r = 5d_n$, $N^* = 2^{10} - 1 = 1023$, and $j = 3$. In this figure, both the packet frame interarrival times and the cell residence times are exponentially distributed. In Fig. 5(a), we observe that CFS outperforms both BFS and NFS in terms of $d_{s,j}$ delay. When λ/μ is small (i.e., $\lambda/\mu < 700$), the $d_{s,j}$ costs for the frame synchronization algorithms have the following relationship:

$$d_{s,j}(\text{CFS}) \approx d_{s,j}(\text{BFS}) < d_{s,j}(\text{NFS}).$$

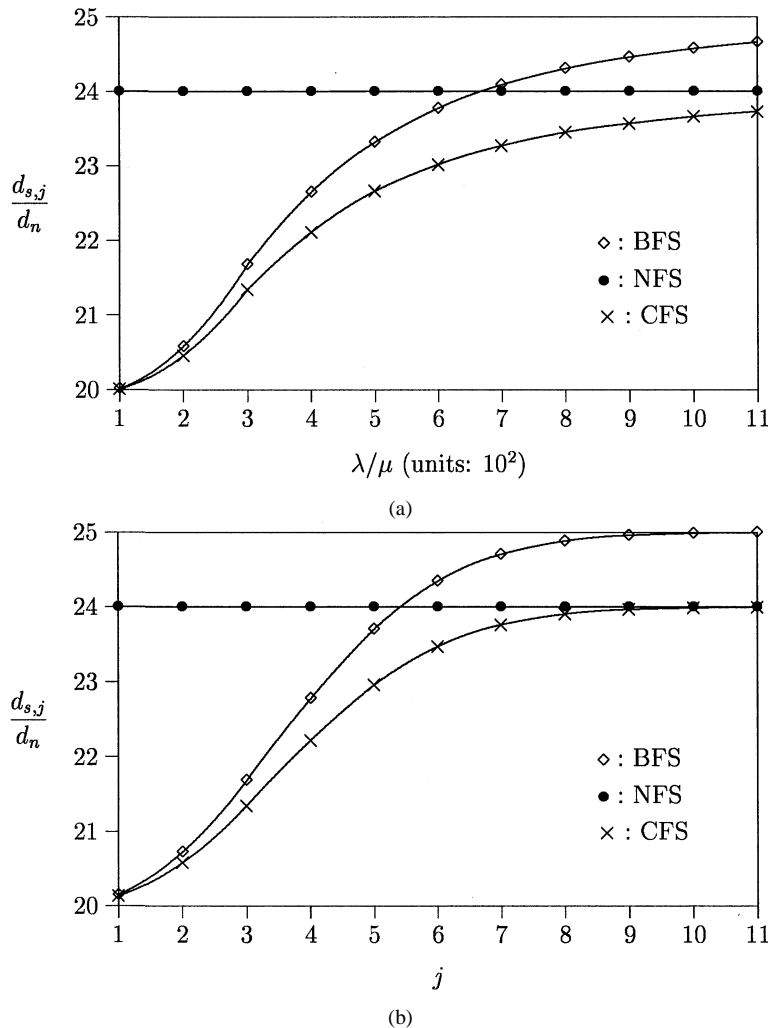


Fig. 5. Effects of λ/μ and j on the frame synchronization algorithms ($d_r = 5d_n$; $N^* = 1023$). (a) Effects of λ/μ ($j = 3$). (b) Effects of j ($\lambda/\mu = 300$).

When $\lambda/\mu = 100$, CFS has 16.63% improvement over NFS in terms of the $d_{s,j}$ performance. As λ/μ becomes large (i.e., $\lambda/\mu \geq 700$), the following relationship is observed:

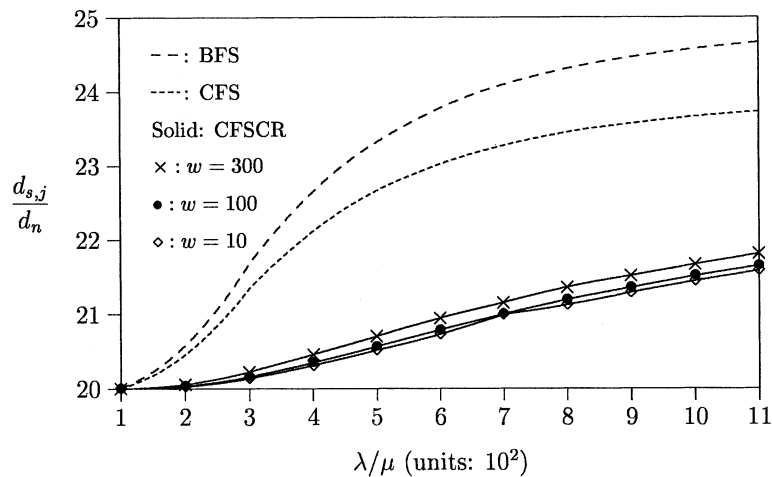
$$d_{s,j}(\text{CFS}) \approx d_{s,j}(\text{NFS}) < d_{s,j}(\text{BFS}).$$

When $\lambda/\mu = 1100$, CFS only has 3.78% improvement over BFS in terms of the $d_{s,j}$ performance. It is clear that the improvement of CFS over BFS is not significant. We will show later (see Fig. 6) that by modifying CFS with counter reset, the CFSCR algorithm may significantly outperform both CFS and BFS. In Fig. 5(b), when j changes, we observe the same trend shown in Fig. 5(a), which indicates that λ/μ and j have the same effects on BFS, NFS, and CFS.

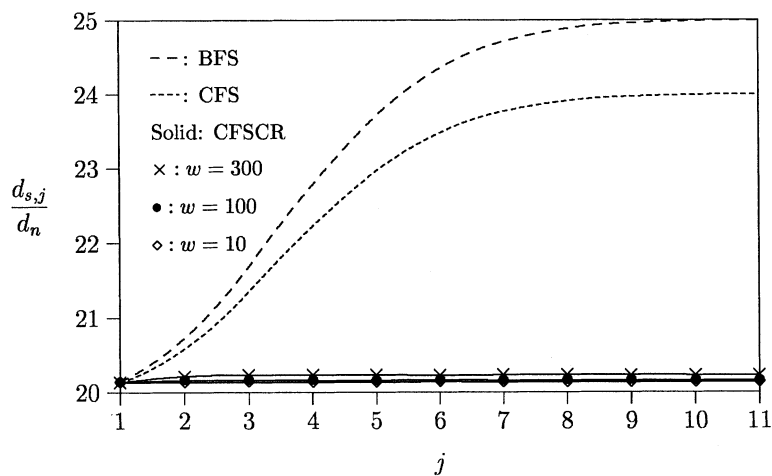
Effects of λ/μ and j on Algorithm CFSCR: Based on the simulation experiments, we investigate the $d_{s,j}$ performance for the CFSCR algorithm. Fig. 6 compares BFS, CFS, and CFSCR with the same input parameter setups as that in Fig. 5. The figure indicates that $d_{s,j}$ (CFSCR) outperforms both $d_{s,j}$ (BFS) and $d_{s,j}$ (CFS) in

all cases. As λ/μ and j become large, the improvements are more significant. For example, when $\lambda/\mu = 11$, CFSCR has 12.5% and 9.05% improvements over BFS and CFS, respectively [see Fig. 6(a)]. When $j = 11$, CFSCR has 19.5% and 16.09% improvements over BFS and CFS, respectively [see Fig. 6(b)]. Fig. 6(a) shows that $d_{s,j}(\text{CFSCR})$ is not affected by j (since CFSCR performs counter reset in every cell crossing). On the other hand, Fig. 6(b) shows that $d_{s,j}(\text{CFSCR})$ increases as λ/μ increases. A larger λ/μ implies that more packet frames are received by the UE in a cell. In this case, it is more likely that for every CFSCR frame synchronization, NFS is exercised and the counters are always reset. Therefore, larger $d_{s,j}$ value is introduced.

Effects of the Window Size w on Algorithm CFSCR: In Fig. 6, we also study the effects of the window size w on the CFSCR algorithm. This figure shows that $d_{s,j}$ (CFSCR) increases slightly as w increases. From (1), it is clear that as w becomes larger, the counters CR_{UE} and CS_s are subtracted by a larger number. Therefore, after switching to a new serving cell, CR_{UE} becomes a smaller value. In this case, it is more likely that BFS is executed when the next cell switching occurs, and smaller $d_{s,j}$ value is expected.



(a)



(b)

Fig. 6. Comparison of BFS, CFS, and CFSCR ($d_r = 5d_n$; $N^* = 1023$). (a) Effects of λ/μ ($j = 3$). (b) Effects of j ($\lambda/\mu = 300$).

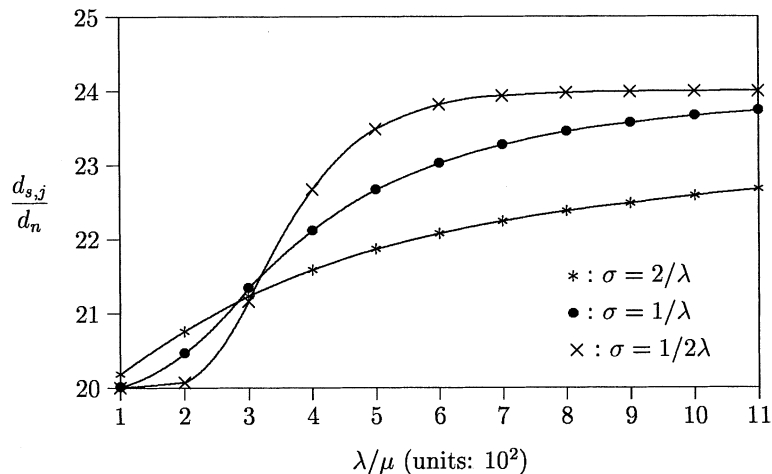


Fig. 7. Effects of cell residence time variance on the CFS algorithm ($d_r = 5d_n$; $N^* = 1023$; $j = 3$).

Effects of Gamma Cell Residence Times: Based on (11) and (14), we consider the Gamma cell residence times with various σ (standard deviation) values. Fig. 7 plots $d_{s,j}$ (CFS) as functions of λ and σ , where the input parameter setups

are the same as that in Fig. 5. The figure indicates that when $\lambda/\mu \leq 300$, $d_{s,j}$ is an increasing function of σ . When $\lambda/\mu > 300$, $d_{s,j}$ is a decreasing function of σ . Note that as σ increases, more long and short cell residence times

TABLE II
THE $d_{s,j}$ VALUES FOR CFS WITH PARETO PACKET FRAME INTERARRIVAL TIMES ($d_r = 5d_n$; $N^* = 1023$; $j = 3$)

λ/μ	100	200	500	800	1100
Exponential	$20.0096d_n$	$20.4583d_n$	$22.665d_n$	$23.45d_n$	$23.7281d_n$
Pareto	$20.0707d_n$	$20.92d_n$	$22.9642d_n$	$23.5466d_n$	$23.7561d_n$
Error	0.3%	2.2%	1.3%	0.41%	0.12%

are observed. Furthermore, the number of small periods is larger than that of long periods (so that the expected value remains the same). We consider two cases:

Case I. $\lambda/\mu < 300$ (i.e., average number of packet frames received by the UE in a cell is less than 300). If σ is small, the UE is expected to receive less than $300 \times 3 \approx N^*$ packet frames during three cell crossings. In this case, the BFS algorithm is executed to deliver frame synchronization messages (with the delay $d_{s,j} = 4d_r$). However, if σ increases (i.e., more longer cell residence times are observed), it becomes likely that the UE receives more than 300 packet frames in a cell, and the NFS algorithm is executed (with the delay $d_{s,j} = 4d_r + 4d_n$). Thus, $d_{s,j}/d_n$ increases as σ increases.

Case II. $\lambda/\mu > 300$ (i.e., the average number of packet frames received by the UE in a cell is larger than 300). If σ is small, the UE is expected to receive more than $900 \approx N^*$ packet frames during three cell crossings. The NFS algorithm is likely to be executed to transmit frame synchronization messages, and the delay $d_{s,j} = 4d_r + 4d_n$ is required. On the other hand, if σ is large (i.e., more short cell residence times are observed), the UE is more likely to receive less than 300 packet frames in a cell, and the BFS algorithm is executed (with the delay $d_{s,j} = 4d_r + 4d_n$). Consequently, $d_{s,j}/d_n$ is a decreasing function of σ .

Effects of Pareto Packet frame Interarrival Times: Based on the simulation experiments, we measure the $d_{s,j}$ values for the CFS algorithm when the packet frame interarrival times have a Pareto distribution with parameters β and l . The β parameter describes the ‘‘heaviness’’ of the tail of the distribution. The Pareto density function is

$$f_P(t) = \left(\frac{\beta}{l}\right) \left(\frac{l}{t}\right)^{\beta+1}$$

and the expected value is

$$E[t] = 1/\lambda = \left(\frac{\beta}{\beta-1}\right)l. \tag{15}$$

If β is between 1 and 2, the variance for the distribution becomes infinity. Once a suitable value for β is selected to describe the traffic characteristics, the l value is determined by the mean of the distribution. It has been shown that the Pareto distribution can approximate the packet traffic very well [7]. Table II lists the $d_{s,j}$ values for both Pareto (where $\beta = 1.2$) and exponential packet frame interarrival times. Other input parameter setups are the same as

that in Fig. 5. In this table, we observe that the $d_{s,j}$ values for Pareto packet frame interarrival times and exponential packet frame interarrival times are roughly the same, which indicates that the selection of exponential packet frame interarrival time distribution is appropriate for primary performance study.

V. CONCLUSION

We proposed an analytic model and simulation experiments to investigate the performance of the frame synchronization algorithms (BFS, NFS, CFS, and CFSCR) for UMTS HSDPA. In BFS, the frame synchronization information is carried through the air interface (i.e., the uplink DPCCCH). In NFS, the frame synchronization information is exchanged between the old and new serving cells through the wireline ATM network. CFS combined BFS and NFS, where the old serving cell decides whether to transmit frame synchronization information through the wireline ATM network or the air interface. CFSCR is an enhancement of CFS. In CFSCR, the counters (that record the number of packet frames processed by the overflow control scheme) are reset to smaller values after every cell switching. Our study indicated that CFSCR outperforms BFS, NFS, and CFS in terms of the cell switching delay $d_{s,j}$. The CFSCR improvements over other frame synchronization algorithms are significant for long HSDPA transmission sessions. When the number of packet frames received by the UE in a cell is small, $d_{s,j}$ increases as the standard deviation of the UE cell residence times increases. On the other hand, if the number of packet frames received by the UE is large, $d_{s,j}$ is a decreasing function of the standard deviation of the cell residence times.

APPENDIX

ABBREVIATIONS AND NOTATIONS

The abbreviations used in this paper are listed as follows

- BFS: Basic Frame Synchronization
- CFS: Combined BFS and NFS
- CFSCR: CFS with Counter Reset
- DPCCCH: Dedicated Physical Control Channel
- FCS: Fast Cell Selection
- HSDPA: High-Speed Downlink Packet Access
- NFS: Network Frame Synchronization
- RNC: Radio Network Controller
- SAW-Hybrid ARQ: Stop-And-Wait Hybrid ARQ
- UE: User Equipment
- UMTS: Universal Mobile Telecommunication System
- UTRAN: UMTS Terrestrial Radio Access Network.

The notations used in our study include

- b : the number of bits used to keep the frame synchronization information in a uplink DPCCCH transmission
- B_{UE} : the number of bits required to carry the CR_{UE} value

- $Cell_i$ ($Cell_s$): a cell in the active set (the serving cell in the active set)
- $Cell_o$ ($Cell_n$): the old serving cell (the new serving cell)
- CR_{UE} : a counter maintained by the UE to indicate the number of received packet frames by the UE
- CS_i (CS_s): a counter maintained by $Cell_i$ ($Cell_s$) to count the number of packet frames that have been processed by $Cell_i$ (that have been received by the UE)
- $d_{s,j}$: the net cost for frame synchronization and cell switching to the $j + 1$ st cell
- d_r (d_n): the expected transmission delays in the air interface (in the ATM network)
- $f^*(s)$: the Laplace transform of $f(t)$
- $f(t)$: the density function of the serving cell residence time of a UE
- J : the number of frame synchronization performed in a HSDPA transmission session
- M : the number of uplink DPCCH transmissions when the UE switches from the j th cell to the $j + 1$ th cell
- N^* : the maximum number that can be carried through one uplink DPCCH transmission
- $n_{r,j}$ ($n_{n,j}$): the number of control messages delivered in the air interface (in the ATM network) when the UE switches from the j th cell to the $j + 1$ th cell during a communication session
- $\Pr[K = k, J = j]$: the probability that the transmission session is not complete at the j th serving cell, and during t_j , there are k packet frames received by the UE
- $\Pr[M = m, J = j]$: the probability that there are m uplink DPCCH transmission when the UE switches to the $j + 1$ st cell
- $r_j^*(s)$: the Laplace transform of $r_j(t)$
- $r_j(t)$: the density function of the session age t_j
- t : the serving cell residence time of a UE
- t_j : the time interval between the start of down-link transmission and when the transmission is switched to the $j + 1$ th serving cell
- w : the window size used in the window-based flow control algorithm between the RNC and the cells in the active set
- β and l : two parameters for the Pareto packet frame inter-arrival time distribution
- λ : the packet frame arrival rate to the UE in a downlink transmission session
- $1/\mu$: the expected cell residence time of a UE
- θ : the number used in the CFSCR scheme, with which the CR_{UE} is subtracted after the UE switches to a new serving cell

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers. Their comments have significantly improved the quality of this paper.

REFERENCES

- [1] 3GPP, "Working Group 2; Radio Interface Protocol Architecture," 3rd Generation Partnership Project; Technical Specification Group Radio Access Network, Tech. Spec. 3G TS 25.301 version 3.4.0 (2000–03), 2000.
- [2] 3GPP, "Technical Specification Group Radio Access Network; High Speed Downlink Packet Access; Overall UTRAN Description; Release 5," 3rd Generation Partnership Project, Tech. Rep. 3G TR 25.855 version 5.0.0 (2001–09), 2001.
- [3] 3GPP, "Physical Layer Aspects of UTRA High Speed Downlink Packet Access; Release 4," 3rd Generation Partnership Project Technical Specification Group Radio Access Network, Tech. Rep. 3G TR 25.848 version 4.0.0 (2001–03), 2001.
- [4] 3GPP, "UTRA High Speed Downlink Packet Access; Release 4," 3rd Generation Partnership Project; Technical Specification Group Radio Access Network, Tech. Rep. 3G TR 25.950 version 4.0.0 (2001–03), 2001.
- [5] 3GPP, "UTRAN Iub Interface: General Aspects and Principles; Release 4," 3rd Generation Partnership Project; Technical Specification Group Radio Access Network, Tech. Spec. 3G TS 25.430 version 4.1.0 (2001–06), 2001.
- [6] 3GPP, "General Packet Radio Service (GPRS); Service Description; Stage 2," 3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects, Tech. Spec. 3G TS 23.060 version 4.1.0 (2001–06), 2001.
- [7] M. Cheng and L.-F. Chang, "Wireless dynamic channel assignment performance under packet data traffic," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1257–1269, July 1999.
- [8] I. Chlamtac, Y. Fang, and H. Zeng, "Call blocking analysis for PCS networks under general cell residence time," in *Proc. IEEE WCNC*, New Orleans, LA, Sept. 1999.
- [9] A. Das, F. Khan, A. Sampath, and H.-J. Su, "Performance of hybrid ARQ for high speed downlink packet access in UMTS," in *Proc. IEEE VTC2001—Fall*, vol. 4, Fall 2001, pp. 2133–2137.
- [10] Y. Fang and I. Chlamtac, "Teletraffic analysis and mobility modeling for PCS networks," *IEEE Trans. Commun.*, vol. 47, pp. 1062–1072, July 1999.
- [11] Y. Fang, I. Chlamtac, and H.-B. Fei, "Analytical results for optimal choice of location update interval for mobility database failure restoration in PCS networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 11, pp. 615–624, June 2000.
- [12] P. Frenger, S. Parkvall, and E. Dahlman, "Performance comparison of HARQ with chase combining and incremental redundancy for HSDPA," in *Proc. IEEE VTC2001—Fall*, vol. 3, Fall 2001, pp. 1829–1833.
- [13] H. Honkasalo, K. Pehkonen, M. T. Niemi, and A. T. Leino, "WCDMA and WLAN for 3G and beyond," *IEEE Wireless Commun.*, vol. 9, pp. 14–19, Apr. 2002.
- [14] W. S. Jeon, D. G. Jeong, and B. Kim, "Design of packet transmission scheduler for high speed downlink packet access systems," in *Proc. IEEE VTC2002—Spring*, vol. 3, 2002, pp. 1125–1129.
- [15] N. L. Johnson, *Continuous Univariate Distributions—1*. New York: Wiley, 1969.
- [16] T. Kawamura, K. Higuchi, Y. Kishiyama, and M. Sawahashi, "Comparison between multipath interference canceller and chip equalizer in HSDPA in multipath channel," in *Proc. IEEE VTC2002—Spring*, vol. 1, 2002, pp. 459–463.
- [17] P. Lin, Y.-B. Lin, and I. Chlamtac, "Overflow control for UMTS high-speed downlink packet access," *IEEE Trans. Wireless Commun.*, submitted for publication.
- [18] Y.-B. Lin, "Reducing location update cost in a PCS network," *IEEE/ACM Trans. Networking*, vol. 5, Jan. 1997.
- [19] R. Love, A. Ghosh, R. Nikides, L. Jalloul, M. Cudak, and B. Classon, "High speed downlink packet access performance," in *Proc. IEEE VTC2001—Spring*, vol. 3, 2001, pp. 2234–2238.
- [20] Lucent, "ARQ technique for HSDPA," Lucent, Tech. Rep. R2A010021.



Phone Lin (M'01) received the B.Sc.S.I.E. degree and the Ph.D. degree from National Chiao Tung University, Taiwan, R.O.C., in 1996 and 2001, respectively.

In 2001, he was appointed as an Assistant Professor in the Department of Computer Science and Information Engineering (CSIE), National Taiwan University, Taipei, R.O.C. His current research interests include personal communications services, wireless Internet, and performance modeling.



Yi-Bing Lin (SM'96–M'96–F'03) received the B.S.E.E. degree from National Cheng Kung University in 1983, and the Ph.D. degree in computer science from the University of Washington, Seattle, in 1990.

From 1990 to 1995, he was with the Applied Research Area at Bell Communications Research (Bellcore), Morristown, NJ. In 1995, he was appointed Professor in the Department of Computer Science and Information Engineering (CSIE), National Chiao Tung University (NCTU), Taiwan,

R.O.C. In 1996, he was appointed Deputy Director of Microelectronics and Information Systems Research Center, NCTU. During 1997–1999, he served as Chairman of CSIE, NCTU. His current research interests include design and analysis of personal communications services network, mobile computing, distributed simulation, and performance modeling. He has published over 150 journal articles and more than 200 conference papers.

Dr. Lin is an Associate Editor of IEEE NETWORK, an Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, an Associate Editor of IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, an Associate Editor of *IEEE Communications Survey and Tutorials*, an Editor of IEEE WIRELESS COMMUNICATIONS MAGAZINE, an Editor of *Computer Networks*, an Area Editor of *ACM Mobile Computing and Communication Review*, a columnist for *ACM Simulation Digest*, an Editor of *International Journal of Communications Systems*, an Editor of *ACM/Baltzer Wireless Networks*, an Editor of *Computer Simulation Modeling and Analysis*, an Editor of *Journal of Information Science and Engineering*, Program Chair for the 8th Workshop on Distributed and Parallel Simulation, General Chair for the 9th Workshop on Distributed and Parallel Simulation, Program Chair for the 2nd International Mobile Computing Conference, Guest Editor for the *ACM/Baltzer MONET* special issue on Personal Communications, a Guest Editor for IEEE TRANSACTIONS ON COMPUTERS Special Issue on Mobile Computing, a Guest Editor for IEEE TRANSACTIONS ON COMPUTERS Special Issue on Wireless Internet, and a Guest Editor for IEEE COMMUNICATIONS MAGAZINE Special Issue on Active, Programmable, and Mobile Code Networking. He is the author of the book *Wireless and Mobile Network Architecture* (coauthored with Imrich Chlamtac; published by Wiley). He received the 1998, 2000, and 2002 Outstanding Research Awards from National Science Council, R.O.C., and 1998 Outstanding Youth Electrical Engineer Award from CIEE, R.O.C. He also received the NCTU Outstanding Teaching Award in 2002. He is an Adjunct Research Fellow of Academia Sinica, and is Chair Professor of Providence University. He serves as consultant of many telecommunications companies including FarEasTone and Chung Hwa Telecom.



Imrich Chlamtac (M'86–SM'86–F'93) received the Ph.D. degree in computer science from the University of Minnesota, Minneapolis.

Since 1997, he has been the Distinguished Chair in Telecommunications at the University of Texas at Dallas, Richardson. He also holds the titles of the Sackler Professor at Tel-Aviv University, Israel, The Bruno Kessler Honorary Professor at the University of Trento, Italy, where he is currently on sabbatical, and University Professor at the Technical University of Budapest, Hungary. He published close to 300

papers in refereed journals and conferences, and is the coauthor of the first textbook on *Local Area Networks* (Lexington Books, 1981,1982,1984) and of *Mobile and Wireless Networks Protocols and Services* (Wiley, 2000); an IEEE NETWORK magazine's 2000 Editor's Choice. He serves as the Founding Editor in Chief of the *ACM/URSI/Kluwer Wireless Networks* (WINET), the *ACM/Kluwer Mobile Networks and Applications* (MONET) journals, and the *SPIE/Kluwer Optical Networks Magazine* (ONM).

Dr. Chlamtac is a Fellow of the ACM, a Fulbright Scholar, and an IEEE Distinguished Lecturer. He is the winner of the 2001 ACM Sigmobile annual award and the IEEE ComSoc TCPC 2002 Award for contributions to wireless and mobile networks, and of multiple best paper awards in wireless and optical networks.