# Credit Allocation for UMTS Prepaid Service

Phone Lin, *Member, IEEE*, Yi-Bing Lin, *Fellow, IEEE*, Chai-Hien Gan, and Jeu-Yih Jeng

***Abstract*—Prepaid phone service requires a user to make payment before calling. In 2.5G or the 3G networks, a user may be engaged in multiple voice and/or data prepaid sessions at the same time. For such services, it is important to distribute appropriate amounts of prepaid credit units to simultaneously executed sessions of a user. By considering the universal mobile telecommunication system (UMTS) as an example, this paper studies the prepaid credit allocation for the prepaid sessions. To simulataneously accommodate more prepaid sessions for a user, we propose a credit reclaim mechanism called prepaid credit reclaim (PCR). Analysis and simulation experiments are conducted to investigate the performance of our mechanism. Our study indicates that PCR can significantly improve the performance of the prepaid mechanism.**

***Index Terms*—Credit allocation, credit reclaim, prepaid services, universal mobile telecommunication system (UMTS).**

## I. Introduction

PREPAID telecommunication service requires a user to make payment before accessing the service. In the second-generation (2G) mobile networks such as global system for mobile communications (GSM), prepaid voice service is implemented as a circuit-switched (CS) domain service [6]. In the 2.5G (e.g., general packet radio service GPRS) or the third-generation (3G), e.g., universal mobile telecommunication system or (UMTS) networks, prepaid data service is also offered in the packet-switched (PS) domain [7]. In a typical 2G network, a customer is allowed to make one prepaid call at a time, and the prepaid charging issue is simple. In the 2.5G or the 3G networks, a customer may be engaged in multiple voice and/or data prepaid sessions at the same time. To support such services, it is important to distribute appropriate amount of prepaid credit to the voice calls and the data sessions that are simultaneously connected to a user. This task is not trivial. The network may assign too many prepaid credit units to the already executed prepaid sessions of the user, and therefore, no prepaid credit units are left for new prepaid sessions requested by the same user. On the other hand, the network may assign insufficient amounts of credit to the prepaid sessions, which results in heavy network traffic for prepaid-related signaling.

Two studies [8], [9] have addressed the prepaid service. These studies focused on the service-node-based prepaid service. In this paper, by using UMTS an intelligent network (IN) approach as an example, we study the prepaid credit allocation so that more prepaid sessions can be simultaneously accommodated for a user without incurring heavy network signaling. Also, a prepaid credit reclaim (PCR) procedure is proposed to re-distibute credit for new prepaid sessions. With PCR, the network can flexibly allocate the prepaid credit units to simultaneously connected sessions, and thus, more prepaid sessions can be accommodated for a user.

Fig. 1 illustrates the simplified UMTS network architecture [1]. In this architecture, a user equipment (UE) accesses the UMTS network through the UMTS terrestrial radio access network [UTRAN; Fig. 1(1)], where the WCDMA technology is adopted in the air interface Uu [2]. In the core network [Fig. 1(2)], the mobility database visitor location register [VLR; Fig. 1(3)] maintains the location information for UEs to access the CS-domain services. The home location register [HLR; Fig. 1(4)] maintains the location information and the subscriber-related data for UEs. The mobile switching center [MSC; Fig. 1(5)] connects to the public switched telephone network (PSTN) to provide the CS-domain services. Serving GPRS support node [SGSN; Fig. 1(6)] and gateway GPRS support node [GGSN; Fig. 1(7)] interact through an IP-based GPRS backbone network to provide the PS-domain services. The SGSN is responsible for delivering packets between a UE and the GGSN. The GGSN provides interworking between the UMTS network and the external data network.

GPP TS23.078 describes an IN approach called customized application for mobile network enhanced logic (CAMEL) [3], [4] that supports UMTS prepaid services. In this approach, three IN functional entities [i.e., gsmSSF, gprsSSF, and gsmSCF; see Fig. 1(8)–(10)] are deployed in the MSC, the SGSN, and the CAMEL GSM-SCF, respectively. The CAMEL GSM-SCF (i.e., the gsmSCF entity) instructs the SGSN/MSC to initiate, terminate, suspend or resume an on going PS/CS call. The MSC (i.e., the gsmSSF entity) and the SGSN (i.e., the gprsSSF entity) communicate with the CAMEL GSM-SCF using the camel application part (CAP) protocol [4]. To set up a CS-domain or a PS-domain call, a CAP dialog is established between the CAMEL GSM-SCF and the MSC/SGSN. The CAP messages are exchanged in this dialogue. The MSC/SGSN is responsible for monitoring the call-related events for a CS/PS call (e.g., CS call setup, Call termination, PDP context activation, and PDP context deactivation [5]) and forwarding

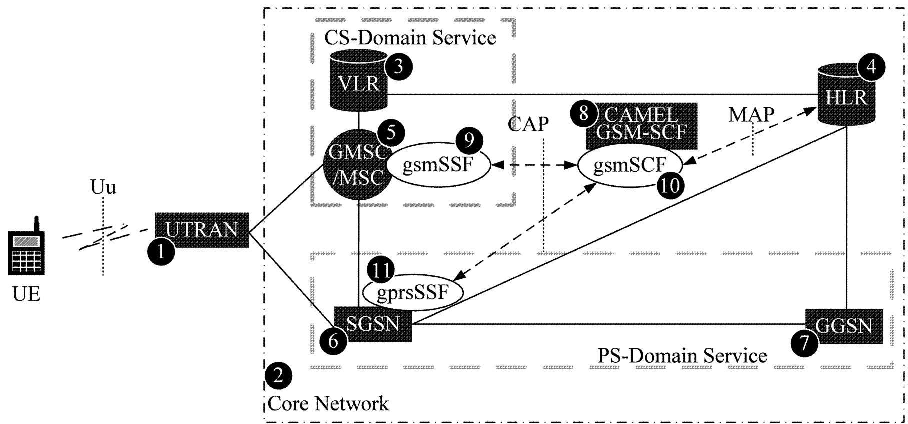
Manuscript received January 10, 2005; revised May 14, 2005 and June 23, 2005. P. Lin's work was sponsored in part by the National Science Council (NSC), R.O.C., under Contracts NSC 94-2213-E-002-083-, NSC 94-2213-E-002-090-, and NSC 94-2627-E-002-001-, Ministry of Economic Affairs (MOEA), R.O.C., under Contract 93-EC-17-A-05-S1-0017, the Computer and Communications Researches Labs/Industrial Technology Research Institute (CCL/ITRL), Chunghwa Telecom Labs, Microsoft, Telcordia Applied Research Center, Taiwan Network Information Center (TWNIC), and Microsoft Corporation, Taiwan. Y.-B. Lin's work was sponsored in part by NSC Excellence project NSC 94-2752-E-009-005-PAE, NSC 94-2219-E-009-001, NSC 93-2213-E-009-100, and NTP VoIP Project under Grant NSC 94-2219-E-009-002, IIS/Academia Sinica, and ITRI/NCTU Joint Research Center. The review of this paper was coordinated by Prof. V. Leung.

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

Y.-B. Lin and C.-H. Gan are with Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: liny@csie.nctu.edu.tw; chgan@csie.nctu.edu.tw).

J.-Y. Jeng is with the Information Technology Laboratory of Telecommunication Laboratories, Chunghwa Telcom Co., Ltd., R.O.C. (e-mail: Jyjeng@cht.com.tw).

Digital Object Identifier 10.1109/TVT.2005.858190




CAMEL: Customized Application for Mobile Network Enhanced Logic
CAP: CAMEL Application Part
GGSN: Gateway GPRS Support Node
HLR: Home Location Register
MAP: Mobile Application Part
MSC: Mobile Switching Center
SGSN: Serving GPRS Support Node
UE: User Equipment
UTRAN: UMTS Terrestrial Radio Access Network
VLR: Visitor Location Register

Fig. 1. UMTS network architecture for prepaid service.

the detected event to the CAMEL GSM-SCF. According to the event type, the CAMEL GSM-SCF instructs the MSC/SGSN to activate IN applications on a connected CS/PS call. For demonstration purpose, we focus on prepaid service of the PS-domain services.

When a mobile user subscribes to the prepaid service, an amount of prepaid credit is purchased and is maintained in the CAMEL GSM-SCF. When the mobile user originates a prepaid session, the SGSN reports this event to the CAMEL GSM-SCF. The CAMEL GSM-SCF checks if the user has enough credit to support this session. If so, the CAMEL GSM-SCF assigns some prepaid credit units to the SGSN. These credit units are decremented at the SGSN in real time either on the traffic volume or the duration time. After the assigned credit units are consumed, the SGSN may ask for more credit units from the CAMEL GSM-SCF. If the credit at the GSM-SCF is depleted, the prepaid session is forced to terminate. To continue the service, the user needs to recharge his/her prepaid credit by purchasing a top-up card. A typical top-up card looks like a lottery scratch card. When the seal is scratched off, a secret code appears. The mobile user dials a toll-free number and follows the instructions of an interactive voice response (IVR) to input his/her mobile station ISDN number (MSISDN; the mobile phone number) and the secret code. The network will verify the code and refresh the account if the code is valid.

3GPP TS 23.078 [3] does not specify the amount of the prepaid credit the CAMEL GSM-SCF should assign to each prepaid session. Since a customer may be engaged in several prepaid sessions at the same time, it is important to distribute appropriate amounts of prepaid credit to simultaneously connected sessions for a user. Without an intelligent prepaid credit allocation mechanism, the CAMEL GSM-SCF may give too many credit units to the already executed prepaid sessions of the user. If so, no prepaid credit is left for any newly incoming requests of the user. To resolve this issue, we propose a prepaid credit distribution (PCD) algorithm for UMTS prepaid services. PCD dynamically adjusts the credit units allocated to simultaneously executed prepaid sessions of a user. Furthermore, we propose a PCR procedure that can reclaim the credit units of already executed prepaid sessions and assign the reclaimed credit to newly requested prepaid sessions of the same user. With PCR, the network can flexibly assign the prepaid credits to the simultaneously connected prepaid sessions. Thus, more prepaid sessions can be accommodated for a user.

## II. UMTS Prepaid Charging Mechanism

The UMTS prepaid charging mechanism is specified in 3GPP TS 23.078 [3]. Fig. 2 illustrates the prepaid charging message flow for the volume-based PS-domain prepaid services. The CAP prepaid service-related messages are exchanged between the SGSN and the CAMEL GSM-SCF. The message flow is described as follows:

Step 1) When the user starts a new prepaid data session $S$, the SGSN informs the CAMEL GSM-SCF of this event through the EventReportGPRS message.

Step 2) Upon receipt of the EventReportGPRS message, the CAMEL GSM-SCF checks the amount $C_r$ of the remaining credit units of the user and determines if the user is allowed to send data. If $C_r$ is not large enough to support this request, then Steps E1 and E2 in Fig. 2 are executed to terminate the session (to be elaborated later). Otherwise, the CAMEL GSM-SCF assigns $k_1$ credit units to session $S$ by sending a message ApplyChargingGPRS maxTransferredVolumn $= k_1$) to the SGSN, where the parameter maxTransfeered-Volum specifies the amount of the credit assigned to this prepaid session. Then, the CAMEL GSM-SCF decrements the remaining credit by $k_1$ units; i.e., $C_r \leftarrow C_r - k_1$. The amount $k_1$ is determined by the PCD algorithm to be described later.

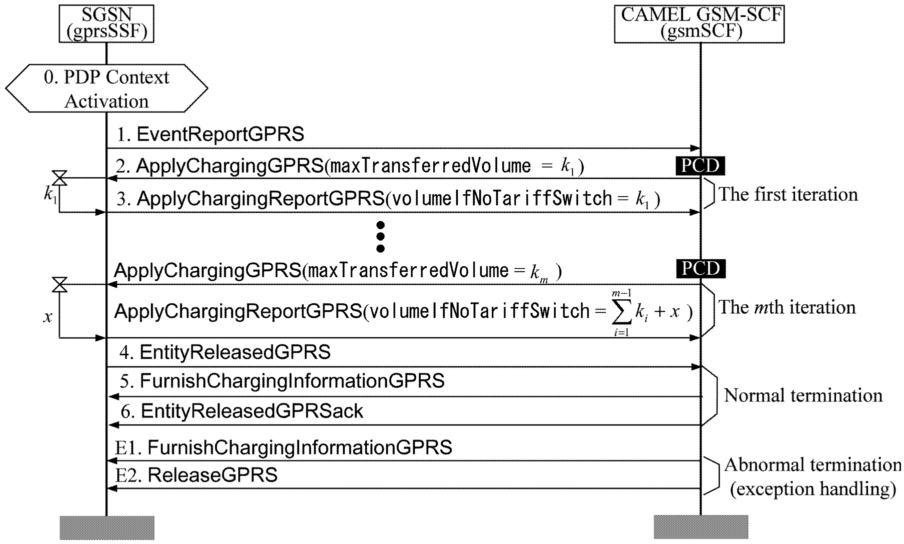



Fig. 2. Message flow for the volume-based prepaid charging.

Step 3) After $k_1$ credit units have been exhausted, the SGSN suspends packet transmission for $S$, and sends a message ApplyChargingReportGPRS volumeIfNoTariffSwitch $= k_1$) to the CAMEL GSM-SCF to indicate that $k_1$ credit units have been consumed, and more credit units are required to serve this session. In this message, the parameter volumeIfNo-TariffSwitch specifies the amount of credit units that have been consumed by $S$ so far. Upon receipt of the ApplyChargingReportGPRS message, the CAMEL GSM-SCF determines that $k_1$ credit units have been consumed.

We define an iteration as a consecutive execution of Steps 2 and 3. Assume that $S$ is normally terminated after $m$ iterations, and that at the $m$th iteration, the SGSN consumes $x$ credit units where $x \leq k_m$. The volumeIfNoTarriffSwitch parameter is set to $\sum_{i=1}^{m-1} k_i + x$ in the $m$th ApplyChargingReportGPRS message. Upon receipt of the $m$th ApplyChargingReportGPRS message, the CAMEL GSM-SCF updates the amount of remaining credit $C_r$ as $C_r \leftarrow C_r + (k_m - x)$. The SGSN completes the session by executing Step 4.

Step 4) By sending the EntityReleasedGPRS message, the SGSN informs the CAMEL GSM-SCF that session $S$ has been terminated.

Step 5) Upon receipt of the EntityReleasedGPRS message, the CAMEL GSM-SCF replies a FurnishCharging-Information message to the SGSN. This message includes the call detailed record (CDR) for this session. The CDR is used to record the PS-domain service usage of the mobile user.

Step 6) The CAMEL GSM-SCF sends the EntityReleasedG-PRSack message to the SGSN, and charging for the session is terminated.

Suppose that after the $i$th iteration, the user does not have enough credit to support the $i + 1$st iteration (i.e., $C_r < k_{i+1}$). Then, the following two steps are executed.

```
Algorithm PCD(θ, n, γ);

begin
        j ← 0;
        while j ≤ n do
                if C_r ≥ θγ^j then
                        return k_i ← θγ^j;
                else j ← j+1;
        return k_i ← NULL;
end
```

Fig. 3. Algorithm PCD.

Step E1) This step is the same as Step 5. The CAMEL GSM-SCF sends a FurnishChargingInformation message that provides the CDR to the SGSN.

Step E2) The CAMEL GSM-SCF sends a ReleaseGPRS message to the SGSN to terminate the ongoing session.

At Step 2, the value $k_1$ may significantly affect the number of simultaneously executed prepaid sessions for a user and the number of the messages exchanged during a prepaid session. If $k_1$ is large, it is more likely to complete a session before $k_1$ credit units are consumed, and a small number of prepaid signaling messages are exchanged. However, if the CAMEL GSM-SCF gives too many credit units to the already executed prepaid sessions of the user, there may not be enough remaining credit units to support new session requests. Therefore, the amounts of credit should be carefully allocated. To flexibly assign the credit units to the prepaid sessions, we propose a prepaid credit distribution (PCD) algorithm. This algorithm is executed at the CAMEL GSM-SCF before the ApplyChargingGPRS message is sent (see the black boxes in Fig. 2). The PCD algorithm is illustrated in Fig. 3. At the begining of Step 2 for the $i$th iteration, the PCD attempts to assign $\theta$ credit units to the requested prepaid session (by setting $j$ to 0). If $C_r > \theta$, then $k_i = \theta$, and the assignment is successful. Otherwise, a reduction factor $\gamma < 1$ is used to reduce the $k_i$ value. Specifically, PCD attempts to find

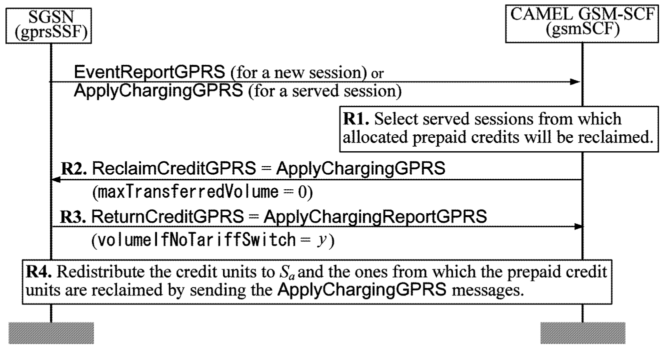



Fig. 4. Message flow for PCR.

the largest $j$ such that $C_r \geq \theta\gamma^j$. Let $n$ be the maximum number of reductions for the $k_i$ value. If $j > n$, then the $k_i$ assignment fails. Otherwise, $k_i$ is set to $\theta\gamma^j$.

Note that if the communication sessions for a mobile user are controlled by the same SGSN, the prepaid credit may be adjusted locally at the SGSN. However, the interactions between the SGSN and the CAMEL GSM-SCF may need to be modified. Furthermore, the simultaneously requested sessions are likely to be issued from different domains (e.g., SGSN, MSC, WLAN gateway, and VoIP call server, etc.), and the prepaid credit units are distributedly decremented by the gateways in these domains. In this case, locally adjusting prepaid within a node does not work. Also, a prepaid account can be shared by several members. In this case, multiple SGSNs (for several authorized members) may simultaneously consume the same prepaid credit source.

## III. Prepaid Credit Reclaim (PCR) Mechanism

In this section, we propose the prepaid credit reclaim (PCR) mechanism. When the execution of PCD fails (i.e., the remaining credit is not large enough to support a new session request or a served prepaid session), this mechanism is triggered to reclaim credit units from the already served sessions. We note that in this scenario, all served sessions (and the new session) are handled by different SGSNs (WLAN gateways or MSCs).

Fig. 4 illustrates the message flow of PCR which consists of four steps. Suppose that the MS makes a new prepaid session request $S_a$ (i.e., an EnventReportGPRS message is received by the CAMEL GSM-SCF) or the UE sends an ApplyChargingReportGPRS message to ask more credit to continue packet transmission for the served prepaid session $S_a$. Assume that $C_r$ at the GSM-SCF is not large enough to support the request, i.e., the execution of PCD fails.

Step R1) This step selects the currently served prepaid sessions, from which the CAMEL GSM-SCF may reclaim credit units to serve the $S_a$ session. Different selection policies, e.g., random or the-largest-first can be adopted. If no served session is found, the PCR mechanism exits, and $S_a$ is rejected. Let $S_1, S_2, S_3, \ldots, S_m$ be prepaid sessions (handled by different SGSNs/MSCs/WLAN gateways; without loss of generality, we use the term SGSN) selected for credit reclaim, where $m$ is the number of the selected sessions.

Step R2) Suppose that Session $S_i$ is handled at SGSN $i (1 \leq i \leq m)$. For every $i$, the CAMEL GSM-SCF sends a ReclaimCreditGPRS message (which can be implemented by the CAP message ApplyChargingGPRS(maxTransferredVolume $=$ 0)) to SGSN $i$. This message informs SGSN $i$ that the assigned credit units of $S_i$ will be reclaimed.

Step R3) Upon receipt of the ReclaimCreditGPRS message, SGSN $i$ suspends packet transmissions for $S_i$, and replies the CAMEL GSM-SCF a ReturnCreditGPRS message (which can be implemented by the CAP message ApplyChargingReportGPRS), where the amount of returned credit units is carried in this message.

Step R4) After the CAMEL GSM-SCF receives all ApplyChargingReportGPRS messages, PCR redistributes the reclaimed credit units to $S_a, S_1, S_2, \ldots$, and $S_m$. Suppose that the amount of the remaining credit is $C_r$. If $(C_r)/(m+1)$ is less than a threshold $\varepsilon (\varepsilon > 0)$, then the reclaimed credit units are returned to $S_1, S_2, \ldots$, and $S_m$ by sending the ApplyChargingGPRS message, the $S_a$ is rejected, and PCR exits. Otherwise (i.e., $(C_r)/(m+1) \geq \varepsilon$), the CAMEL GSM-SCF sends the ApplyChargingGPRS(maxTransferredVolume $=$ $(C_r)/(m+1)$) message to SGSN $i$ for $S_i$.

Two reclaim-related message types are introduced in PCR: ReclaimCreditGPRS and ReturnCreditGPRS. These two message types can be implemented by reusing existing CAP message types, ApplyChargingGPRS and ApplyChargingReportGPRS, respectively (see Steps R1 and R2 in Fig. 4). To trigger PCR, we add reclaim logic functions in the CAMEL GSM-SCF and the SGSN. When the CAMEL GSM-SCF receives an EventReport message and finds that the user does not have enough credit to support this new prepaid session, the reclaim logic in the CAMEL GSM-SCF is triggered to send the ReclaimCreditGPRS message. When the SGSN receives the ReclaimCreditGPRS message, the reclaim logic of the SGSN suspends the current data transmission, and returns the remaining prepaid credit through the ReturnCreditGPRS message. It is clear that the implementation of the PCR procedure does not modify the CAP protocol and only requires minor modifications to the SGSN and the CAMEL GSM-SCF. If the reclaim logic is not implemented in the SGSN, then the ReclaimCreditGPRS message is ignored when SGSN receives this message. In other words, our approach is backward compatibile where credit reclaiming is exercised when the SGSN is equipped with PCR, and the standard 3GPP TS23.078 procedure is exercised if PCR is not installed in the SGSN.

## IV. Performance Evaluation

In this section, we describe simulation experiments to investigate the performance of PCD and PCR. Our simulation model is based on an event-driven approach widely adopted in mobile network studies [12], [13]. The description of the simulation model is given in Appendix.

In this study, we consider two types of prepaid session requests: the streaming type and the interactive type. The prepaid session arrivals for a user form a Poisson process with rate $\Lambda$. When a prepaid session request arrives, the arrival is a streaming prepaid session with probability $\beta$, and the arrival is an interactive session with probability $1 - \beta$.

In a streaming session, the inter-arrival times between two packet arrivals are exponentially distributed with mean $1/\lambda$. In an interactive session, the inter-arrival times form a Pareto distribution with mean $1/\lambda$ and parameters $\upsilon$ and $l$. The shape parameter $\upsilon$ describes the "heaviness" of the tail of the distribution. The scale parameter $l$ determines the mean of the distribution. It has been shown that Pareto distribution appropriately approximates the Internet interactive traffic [10]. The Pareto density function is

$$f_p(x) = \left(\frac{\upsilon}{l}\right)\left(\frac{l}{x}\right)^{\upsilon+1}$$

with the expected value

$$\frac{1}{\lambda} = \left(\frac{\upsilon}{\upsilon - 1}\right) l. \tag{1}$$

If $\upsilon$ is between 1 and 2, the variance for the distribution becomes infinite. Once a suitable value for $\upsilon$ is selected to describe the traffic characteristics, the $l$ value can be derived from the mean of the distribution using (1). To simulate the Internet traffic [10], we set $\upsilon = 1.2$. In either a streaming session or an interactive session, when a packet arrives, the arrival packet ends the session with probability $1 - \alpha$, and the session continues with probability $\alpha$.

To ensure the stability of the simulation results, 100 000 experiments are executed. For each user, we simulate $N_s$ sessions. Initially, the user has $C$ prepaid credit units. Let $M_t$ be the total number of iterations executed in all prepaid sessions of a user, $N_a$ be the total number of prepaid sessions that are accommodated, and $N_c$ be the total number of prepaid sessions that are completely served (i.e., they are not forced to terminate). This study investigates three performance measurements, the expected values $E[(M_t)/(N_a)]$, $E[N_a]$, and $E[N_c]$. The input parameters considered in our study include

- $\lambda$ packet arrival rate in a session;
- $\Lambda$ arrival rate of new session requests;
- $\alpha$ probability that a packet arrival continues the prepaid session;
- $\beta$ probability that an arrival is a streaming packet;
- $C$ initial amount of credit for a user before starting the prepaid service;
- $\theta$ maximum amount of credit units that can be allocated in an iteration;
- $\varepsilon$ a threshold for the redistribution of the remaining credit units;
- $\gamma$ reduction ratio used to scale down the amount of allocated credit;
- $m$ maximum number of served prepaid sessions selected for credit reclaim;
- $n$ maximum number of reductions on the $k_i$ values that can be performed while PCD is exercised for a requested prepaid session; and
- $N_s$ number of prepaid sessions simulated in an experiment.

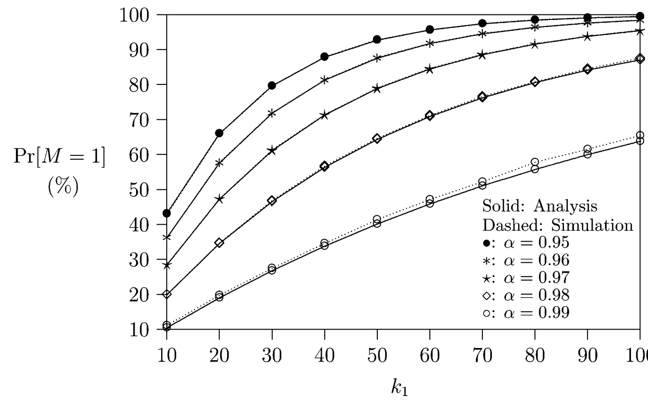



Fig. 5. Validation of simulation and analytical results.

Let $M$ be the number of the iterations required to complete a served prepaid session. Let $k_1$ be the amount of credit units allocated in the first iteration for the prepaid session. To simplify our discussion, each packet transmission consumes one credit unit. If the number of packet arrivals in a session is less than or equal to $k_1$, then only one iteration is required to complete the packet transmission for the session. Thus, the probability that $M = 1$ can be calculated by using the following equation:

$$\Pr[M = 1] = \sum_{j=0}^{k_1} \alpha^j (1 - \alpha) = 1 - \alpha^{k_1+1}. \tag{2}$$

Our simulation model is partially validated by (2). In Fig. 5, the solid curves represent $\Pr[M = 1]$ values based on (2), and the dashed curves are based on simulation experiments. This figure indicates that both analysis and simulation are consistent. In the following, we investigate the effects of $C, n, \theta$, and $\beta$ on the performances of PCD and PCR based on the simulation experiments. Then we consider the effects of the variance of intersession arrival times. In our study, we set $\Lambda = \lambda/5, \alpha = 0.95, m = 1$, and $0 \leq n \leq 3$. For other parameter setups, we observed the similar results, which are not shown in this paper. The setups for $C, \beta$, and $\theta$ are given in the caption of each figure.

Effects of $n$ and $C$: Fig. 6 plots $E[N_a], E[N_c]$, and $E[(M_t)/(N_a)]$ as functions of total credit units $C$ and $n$. The figure indicates that as $n$ increases, $E[N_a]$ significantly increases [Fig. 6(a)], $E[N_c]$ slightly increases [Fig. 6(b)], but $E[(M_t)/(N_a)]$ drops significantly [Fig. 6(c)]. When $n = 0$, the CAMEL GSM-SCF always attempts to allocate $\theta$ credit units in each iteration. As $n$ increases, the CAMEL GSM-SCF can accommodate more new prepaid session requests (by allocating smaller $k_i$ credit units), and $E[N_a]$ significantly increases. For the same reason, more served prepaid sessions are complete; that is, this factor increases $N_c$. However, when more sessions are accommodated, the credit units are more likely to be consumed fast, and the served sessions have less chance to be complete; that is, this factor

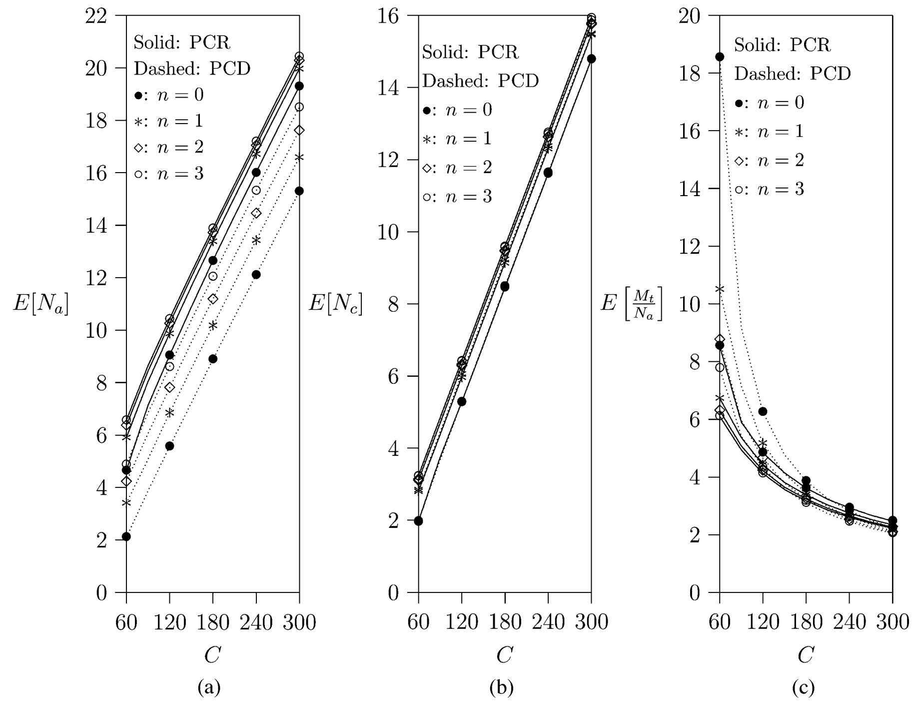



Fig. 6. Effects of $n$ and $C(\gamma = 0.5; \Lambda = \lambda/5; \alpha = 0.95; \beta = 0.5; \theta = 40; N_s = 30; m = 1; \varepsilon = 1)$. (a) $E[N_a]$. (b) $E[N_a]$. (c) $E[\frac{M_1}{N_a}]$.

decreases $N_c$. The net effect of the above two conflicting factors results in slightly increasing $E[N_c]$ values as $n$ increases.

In Fig. 6(c), as $n$ increases, $E[(M_t)/(N_a)]$ decreases. This phenomenon is significant when $C$ is small. For each session request (either accepted or blocked), at least one iteration is required for prepaid credit request. Let $x$ be the expected number of iterations required for an accepted prepaid session. The total number $M_t$ of iterations for the $N_s$ prepaid sessions can be approximately computed by using the following equation:

$$M_t \approx N_s + (x-1)E[N_a]$$
$$\Rightarrow \frac{M_t}{E[N_a]} \approx \frac{N_s}{E[N_a]} + x - 1 \quad (3)$$

To interpret (3), we consider the following two phenomena:

Phenomenon 1) When $C$ is small, $E[N_a]$ increases significantly as $n$ increases [see Fig. 6(a)].

Phenomenon 2) In Fig. 6, we set $\theta = 40$ and $\gamma = 0.5$. When $C = 60$ (i.e., $C$ is small), and $n = 0$, for a served prepaid session, at least one iteration is required. For $n = 1$, we have $k_i \geq \theta\gamma^n = 20$. Since the sum of the total allocated credit units are no larger than $C$, that is, $xk_i \leq 60$, we have $x \leq 3$. Therefore, for PCD, when $n$ is changed from 0 to 1, $x$ is at most increased by two.

Observe the above two phenomena in (3), with small $C$, since $E[N_a]$ significantly increases, and $x$ slightly increases, $E[(M_t)/(N_a)]$ decreases as $n$ increases. Fig. 6(c) indicates that when $C$ is large (e.g., $C = 300$), for different $n$ setups, the $E[(M_t)/(N_a)]$ values are almost identical. When $C$ is large, for most credit requests, the network can satisfy these requests by allocating $\theta$ credit units to them. Therefore, $n$ insignificantly affects $E](M_t)/(N_a)]$.

This figure also shows that for PCR, $E[N_a], E[N_c]$, and $E[(M_t)/(N_a)]$ do not change significantly when $n \geq 2$. Since PCR can reclaim credit, it is not sensitive to the change of $n$.

Effects of $\theta$: Fig. 7 shows the effects of $\theta$ on $E[N_a], E[N_c]$, and $E](M_t)/(N_a)]$. The figure indicates that when $\theta$ increases, $E[N_a]$ and $E[N_c]$ decrease, and $E[(M_t)/(N_a)]$ increases. When $\theta$ increases, more credit units are required to accept a request. Therefore, a request has less chance to be served, and thus, $E[N_a]$ and $E[N_c]$ decrease. For $E[(M_t)/(N_a)]$, since $E[N_a]$ drops significantly as $\theta$ increases, large $E[(M_t)/(N_a)]$ values are observed according to (3).

When $\theta$ increases, the decrease of $E[N_a]$ for PCR is less significant than that for PCD. Its result is due to the fact that the reclaim mechanism can flexibly reclaim credit units when the remaining credit units at the GSM-SCF are not large enough to support a new request.

Effects of $\beta$: Fig. 8 examines the effects of mixed traffics by changing the $\beta$ value. A larger $\beta$ implies that more requests are interactive prepaid sessions. Fig. 8 shows that $E[M_t], E[N_a]$, and $E[N_c]$ are insignificantly affected by $\beta$. In other words, the prepaid credit allocation is insignificantly affected by the traffic pattern of sessions.

Effects of the Variation of the Distributions for Session Interarrival Times: We assume that the session interarrival times for a user have the Gamma distribution with mean $1/\Lambda$ and variance $\upsilon_s = (1)/(\Lambda^2\omega)$, where $\omega$ is the shape parameter. Gamma distributions are considered because they can be used to approximate many other distributions [11]. In Fig. 9, we observe the following:

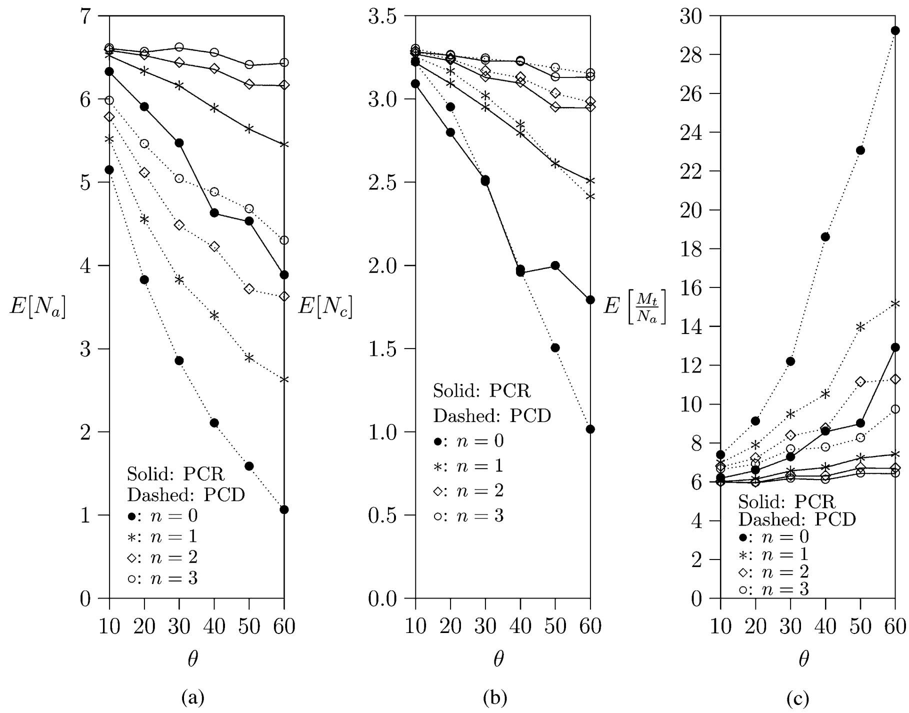



Fig. 7. Effects of $\theta(C = 60; \gamma = 0.5; \Lambda = \lambda/5; \alpha = 0.95; \beta = 0.5; N_s = 30; m = 1; \varepsilon = 1)$. (a) $E[N_a]$. (b) $E[N_a]$. (c) $E[\frac{M_1}{N_a}]$.

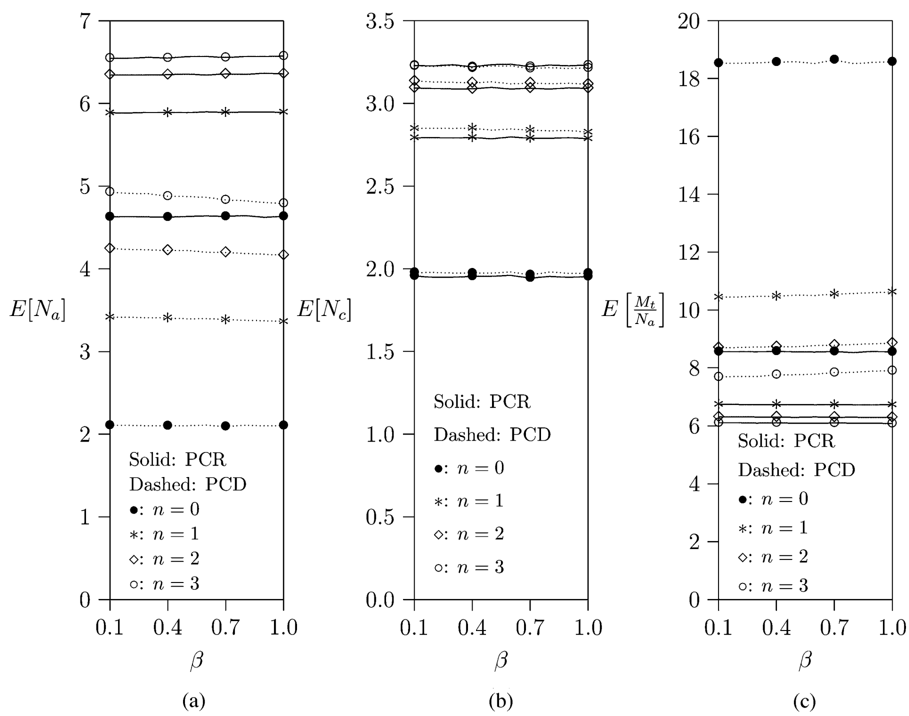



Fig. 8. Effects of $\beta(C = 60; \gamma = 0.5; \Lambda = \lambda/5; \alpha = 0.95; \theta = 40; N_s = 30; m = 1; \varepsilon = 1)$. (a) $E[N_a]$. (b) $E[N_a]$. (c) $E[\frac{M_1}{N_a}]$.

1) When $\upsilon_s < (1)/(\Lambda^2)$, $E[N_a], E[N_c]$, and $E[(M_t)/(N_a)]$ are not significantly affected.
2) For PCD, when $\upsilon_s \geq (10)/(\Lambda^2)$, $E[N_a]$ and $E[N_c]$ decrease, and $E[(M_t)/(N_a)]$ increases as $\upsilon_s$ increases. In other words, the performance of PCD becomes worse as $\upsilon_s$ increases.
3) For PCR, when $\upsilon_s \geq (1)/(\Lambda^2), E[N_a]$ increases significantly, $E[N_c]$ increases slightly, and $E[(M_t)/(N_a)]$ increases slightly as $\upsilon_s$ increases. In other words, with larger $\upsilon_s$, the performance of PCR improves.

The preceding observations are explained as follows. For a large variance $\upsilon_s$, more small session inter-arrival times

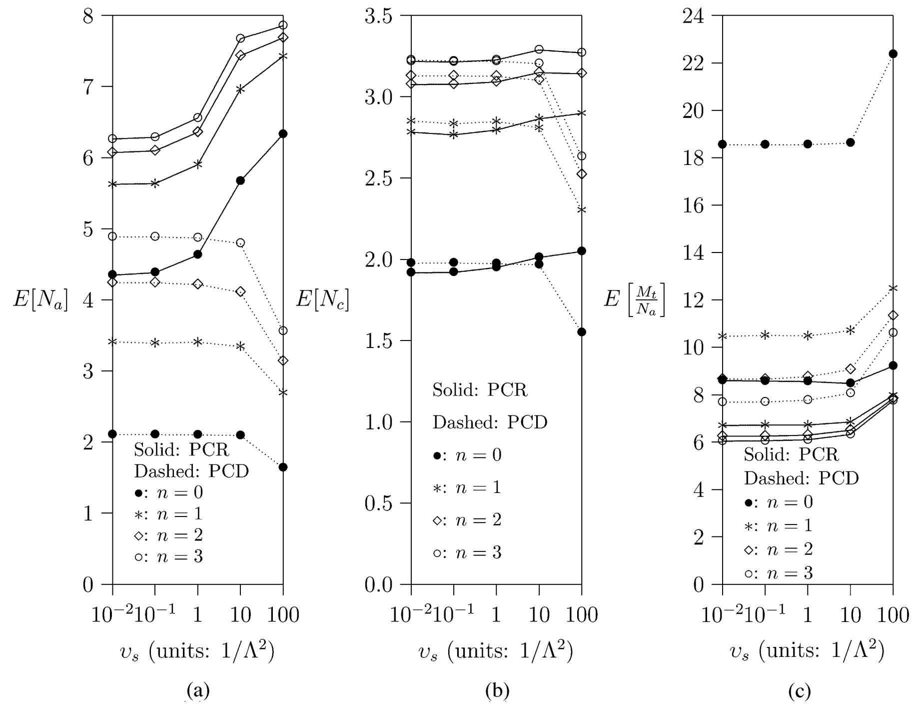



Fig. 9. Effects of the variances of session inter-arrival times ($C = 60; \gamma = 0.5; \beta = 0.5; \Lambda = \lambda/5; \alpha = 0.95; \theta = 40; N_s = 30; \varepsilon = 1$). (a) $E[N_a]$. (b) $E[N_a]$. (c) $E[\frac{M_1}{N_a}]$.

are observed. Thus, more prepaid session requests arrive in a short period, and all credit units are likely to be assigned in a short time period. Also more session arrivals are observed during the service periods of the served prepaid sessions. For PCD, it is more likely that the remaining credit units $C_r$ is insufficient to accommodate newly incoming requests. Thus, $E[N_a]$ and $E[N_c]$ decrease, and from (3), $E[(M_t)/(N_a)]$ increases. With PCR, the relaim mechanism can be triggered when the prepaid session arrives during the service periods of the served sessions, and thus we observe that $E[N_a]$ increases significantly.

Effects of Large Traffic: Fig. 10 studies how large traffic of prepaid sessions affects the performances of PCD and PCR by increasing $\Lambda$ (i.e., the prepaid session arrival rates) and $\alpha$ (i.e., the probability that a packet arrival continues the prepaid session). As shown in Fig. 10, as $\alpha$ increases, for PCD and PCR, $E[N_a]$ and $E[N_c]$ decrease, and $E[(M_t)/(N_a)]$ increases. A larger $\alpha$ implies that there are more packet arrivals in a session. More credit units are consumed to complete a session, and more iterations are required for a session.

Fig. 10 also indicates that for both PCD and PCR, as $\Lambda$ increases, $E[N_a]$ and $E[N_c]$ increase, and $E[(M_t)/(N_a)]$ decreases. When $\Lambda$ increases, more prepaid session requests arrive in a short period. The prepaid session requests have better chance to get services before the credit units are exhausted. Thus, we observe larger $E[N_a]$ and $E[N_c]$ values. Then, according to (3), since $E[N_a]$ increases, we have smaller $E[(M_t)/(N_a)]$ values.

Effects of $m$: Fig. 11 investigates the effects of $m$ (i.e., the maximum number of the selected prepaid sessions for credit reclaim) on the performances of PCD and PCR. As shown in Fig. 11, a larger $m$ setup does not improve the $E[N_a]$ and $E[N_c]$ performances but introduces more iterations for credit assignment (i.e., $E[(M_t)/(N_a)]$ increases). Thus, it is preferred to set $m = 1$ for PCR.

Comparing PCD and PCR: From Figs. 6 and 7, PCR is less sensitive to the input parameters (e.g., $n, C, \theta$, and $\beta$) than PCD is. Since PCR can reclaim credit units and re-distribute them to the prepaid sessions, PCR is not significantly affected by the input parameters, and its performance (in terms of $E[N_a], E[N_c]$ and $E[(M_t)/(N_a)]$) is better than PCD.

## V. Concluding Remarks

This paper investigated the prepaid services for the 2.5G or the 3G networks where multiple voice and/or data prepaid sessions are simultaneously supported for a user. We described the prepaid network architecture based on UMTS and proposed a PCD algorithm that dynamically allocates the amount of credit to support simultaneously prepaid sessions of a user. Then, we proposed a PCR mechanism that reclaims credit units from the currently served prepaid sessions. This mechanism provides extra flexibility for credit allocation.

Simulation experiments were conducted to investigate the performances of the PCD and the PCR mechanisms. Our study indicated that with PCR, more prepaid sessions can be simulataneously accommodated for a user as compared with PCD. For PCR, when $n$ (i.e., the maximum number of reductions that can be performed for an iteration in PCD) is larger or equal to two, the performance enhancement becomes insignificant. We also observed that the prepaid credit allocation is

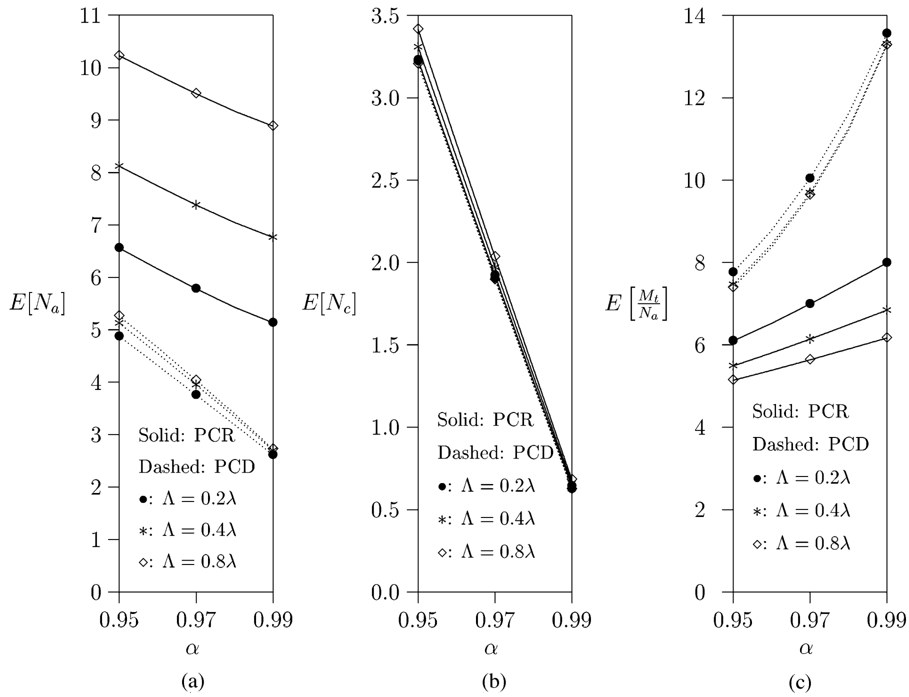



Fig. 10. Effects of the traffic of prepaid sessions ($C = 60; \gamma = 0.5; \beta = 0.5; \theta = 40; n = 3; N_s = 30; m = 1; \varepsilon = 1$). (a) $E[N_a]$. (b) $E[N_a]$. (c) $E[\frac{M_1}{N_a}]$.

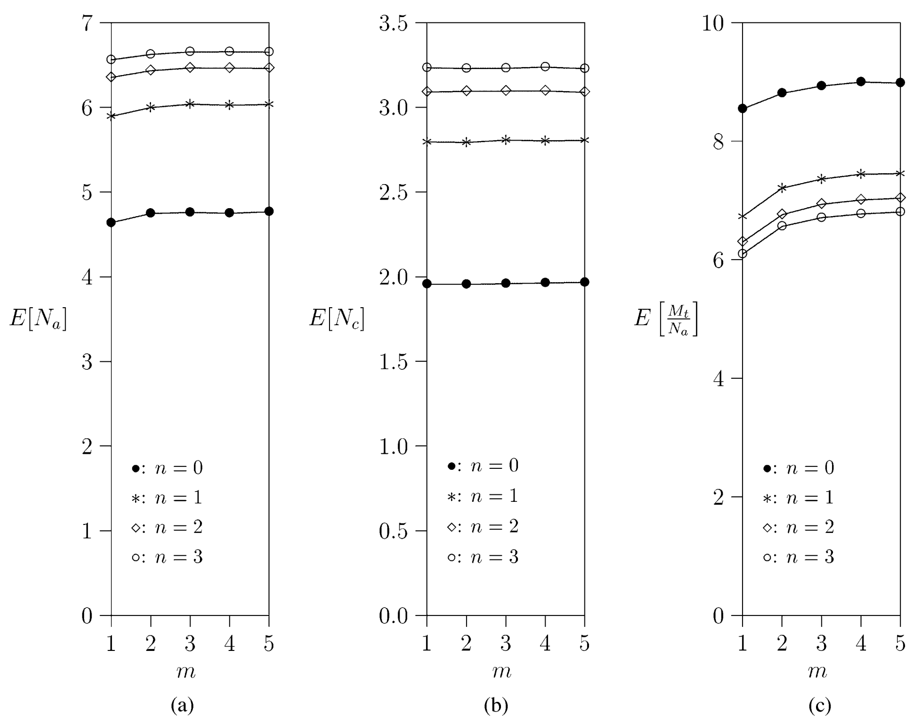



Fig. 11. Effects of $m$ on PCR ($C = 60; \gamma = 0.5; \Lambda = \lambda/5; \alpha = 0.95; \beta = 0.5; \theta = 40; N_s = 30; \varepsilon = 1$).

independent of the traffic patterns. PCR has better performance than PCD when variance of the session interarival times is larger.

As a final remark, PCR is a pending patent issued from Chung-Hua Telecom.

## Appendix

## Simulation Model for the PCR Mechanism

This Appendix describes the discrete-event simulation model for the PCR mechanism. The simulation defines three types of events, session_arrival, session_complete, and packet_arrival to represent a new session arrival, the completion of a session, and a packet arrival for a session, respectively. The events are inserted into an event list and are deleted/processed from the list in the nondecreasing timestamp order. Another list, named serving list, maintains the status for the currently served prepaid sessions. For a session_arrival event, if the request is granted for service, a corrsponding record for this request is generated and inserted into the serving list.

A simulation clock $t_s$ is maintained to indicate the progress of the simulation. Three variables are used to calculate the output measures $E[N_a], E[N_c]$, and $E[(M_t)/(N_a)]$, including

- $N_a$ number of total prepaid session arrivals accommodated for a user;
- $N_c$ number of total sessions completed for a user; and
- $M_t$ the number of the iterations (exchange of Steps 2 and 3 in Fig. 2) executed before the total credit units are exhausted.

We repeat the simulation runs for $K$ times and compute the outputs as follows:

$$E[N_a] = \frac{\sum_{i=1}^{K} N_a[i]}{K}$$

$$E[N_c] = \frac{\sum_{i=1}^{K} N_c[i]}{K}$$

and

$$E\left[\frac{M_t}{N_a}\right] = \frac{\sum_{i=1}^{K} \frac{M_t[i]}{N_a[i]}}{K}$$

where $N_a[i], N_c[i]$, and $M_t[i]$ are the $N_a$, $N_c$, and $M_t$ values in the $i$th simulation run. The following variables are also used in the simulation:

- $C_r$ amount of the remaining credit units for a user;
- $C_s$ amount of the remaining credit units allocated in an iteration for Session $s$;
- $k_i$ amount of credit units allocated in the $i$th iteration for Session $s$;
- $N_t$ number of session_arrival events being executed; and
- $C_e$ amount of credit units reclaimed from the served prepaid sessions.

Fig. 12 illustrates the flowchart of the simulation model. The initial values for input parameters (e.g., $\lambda, \Lambda, \alpha$, etc.) are set up, and the serving list is set to NULL [i.e., the serving list is empty; Step (1)]. A session_arrival event is generated and inserted into the event list [Step (2)], and the total number $N_t$ of session_arrival events is set to one [Step (3)]. We simulate $N_s = 30$ session_arrival events for a user, and repeat the simulation runs for $K = 100\,000$ times to ensure the stability of the simulation results. A loop is executed until the event list is empty [Step (4)]. At Step (5), the next event is retrieved from the event list. Step (6) checks the type of the event. The actions in Steps (7), (32), and (34) are executed to process the session_arrival event, the session_complete event, and the packet_arrival event, respectively.

Step (7) checks whether $N_t$ is less than $N_s$. If so, the next session_arrival event is generated and inserted into the event list [Step (8)]. The $N_t$ value is increased by one [Step (9)]. Step (10)

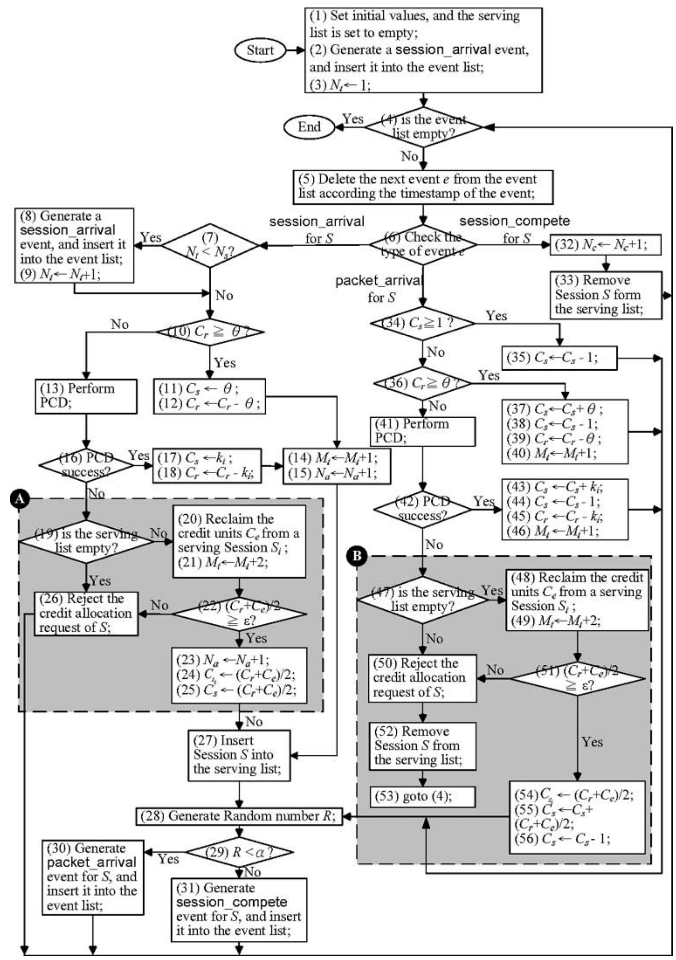



Fig. 12. Flowchart of the simulation model for the PCR mechanism.

checks whether $C_r$ is large enough to accommodate Session $S$. If so (i.e., $C_r \geq \theta$), $\theta$ credit units are allocated to Session $S$, and $C_s$ is set to $\theta$ [Step (11)]. Then $C_r$ is reduced by $\theta$ [Step (12)]. The number of total iterations $M_t$ is increased by one [Step (14)], $N_a$ (i.e., the number of accommodated sessions) is increased by one [Step (15)], and the simulation procceds to Step (27) where Session $S$ is insered into the serving list. Otherwise [i.e., $C_r < \theta$ at Step (10)], the PCD algorithm (described in Fig. 3) is executed [Step (13)]. If PCD successfuly exits, $k_i$ credit units are allocated to Session $S$, and $C_s$ is set to $k_i$ [Step (17)]. Then, $C_r$ is decreased by $k_i$ [Step (18)], $M_t$ is increased by one [Step (14)], $N_a$ is increased by one [Step (15)], and Session $S$ is insered into the serving list [Step (27)]. Otherwise (i.e., PCD fails), the PCR mechanism is executed (see Box A).

Step (19) checks the serving list to see if there is any currently served session. If not, Session $S$ is rejected at Step (26). If so, Step (20) selects the session $S_i$ that has the maximum remaining credits at SGSN $i$, and reclaims $C_e$ credit units from $S_i$. $M_t$ is increased by two [Step (21]; one iteration is for credit reclaim and credit reallocation for $S_i$; the other iteration is for credit request and credit reallocation for $S$). Step (22) checks if $(C_r + C_e)/2 < \varepsilon$. If so, Step (26) rejects the credit allocation request of Session $S$. Otherwise, $N_a$ is increased by one at Step (23), and $(C_r + C_e)/2$ units are allocated to

Sessions $S$ and $S_i$, respectively [Steps (24) and (25)]. Then, Session $S$ is insered into the serving list [Step (27)]. Following the uniform distribution, Step (28) generates a random number $R$ where $0 < R < 1$. Step (29) checks whether $R < \alpha$. If so, Step (30) generates a packet_arrival event for Session $S$. Otherwise (i.e., $R \geq \alpha$), Step (31) generates a session_compete event for Session $S$.

If the event type is session_compete at Step (6), $N_c$ is increased by one [Step (32)], Session $S$ is removed from the serving list [Step (33)], and the simulation proceeds to Step (4).

If the event type is packet_arrival at Step (6), Step (35) checks whether $C_s$ is large enough to support Session $S$. As mentioned previously, our study assumed that each packet transmission consumes one credit unit. If $C_s \geq 1$ (i.e., the remaining credit units are large enough to satisfy the packet transmission), $C_s$ is decreased by one. Then the simulation proceeds to Step (28). Otherwise (i.e., $C_s = 0$), Step (36) checks whether the remaining credit units (i.e., $C_r$) is no less than $\theta$. If so (i.e., $C_r \geq \theta$), the credit units $\theta$ are allocated to Session $S$ [Step (37)], and then $C_s$ is decreased by one [Step (38)]. Then, Step (39) reduces $C_r$ by $\theta$ and increases $M_t$ by one. Otherwise [i.e., $C_r < \theta$ at Step (36)], the PCD algorithm is executed [Step (41)]. If PCD successfully exits, Step (43) allocates the credit units $k_i$ to Session $S$, and $C_s$ is reduced by one [Step (44)]. Then, Step (45) reduces $C_r$ by $k_i$, and Step (46) increases $M_t$ by one. If PCD fails, the PCR mechanism is executed, which is similar to the PCR mechanism for the session_arrival event (see Box B in Fig. 12). If PCR successfully exits, then the simulation proceeds to Step (28). Otherwise, Step (50) rejects the credit allocation request of Session $S$, and Step (52) removes it from the serving list.

## ACKNOWLEDGMENT

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

## REFERENCES

[1] *3GPP. 3rd Generation Partnership Project, Technical Specification Group Services and Systems Aspects, General Packet Radio Service (GPRS), Service Description Stage 2*. Tech. Rep. Tech. Specification 3G TS 23.060 version 4.1.0 (2001–06), 2001.

[2] *3GPP. 3rd Generation Partnership Project, Technical Specification Group Radio Access Network, Radio Interface Protocol Architecture, Release 5*. Tech. Rep. 3G TS 25.301 version 5.2.0 (2002–09), 2002.

[3] *3GPP. 3rd Generation Partnership Project, Technical Specification Group Core Network, Customized Applications for Mobile network Enhanced Logic (CAMEL) Phase 4, Stage 2*. Tech. Rep. 3G TS 23.078 version 5.0.0 (2003–09), 2003.

[4] *3GPP. 3rd Generation Partnership Project, Technical Specification Group Core Network, Customized Application for Mobile Network Enhanced Logic (CAMEL) CAMEL Application Part (CAP) Specification*. Tech. Rep. 3G TS 29.078 version 5.4.0 (2003–06), 2003.

[5] *3GPP. 3rd Generation Partnership Project, Technical Specification Group Services and Systems Aspects, General Packet Radio Service (GPRS), Service Description, Stage 2*. Tech. Rep. 3G TS 23.060 version 6.1.0 (2003–06), 2003.

[6] *3GPP. 3rd Generation Partnership Project, Technical Specification Group Services and System Aspects, Telecommunication Management Charging Management, Charging Data Description for the Circuit Switched (CS) Domain*. Tech. Rep. 3G TS 32.205 version 4.7.0 (2004–03), 2004.

[7] *3GPP. 3rd Generation Partnership Project, Technical Specification Group Services and System Aspects, Telecommunication management, Charging Management, Charging Data Description for the Packet Switched (PS) Domain*. Tech. Rep. 3G TS 32.215 version 4.7.0 (2004–03), 2004.

[8] M.-F. Chang and W.-Z. Yang, "Performance of mobile prepaid and priority call services," *IEEE Commun. Lett.*, vol. 6, no. 2, pp. 61–63, Feb. 2002.

[9] M.-F. Chang, W.-Z. Yang, and Y.-B. Lin, "Performance of service-node-based mobile prepaid service," *IEEE Trans. Veh. Technol.*, vol. 51, no. 3, pp. 597–612, May 2002.

[10] M. Cheng and L.-F. Chang, "Wireless dynamic channel assignment performance under packet data traffic," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 7, pp. 1257–1269, Jul. 1999.

[11] A.-M. Law and W.-D. Kelton, *Simulation Modeling and Analysis*, 3rd ed. New York: McGraw-Hill, 2000.

[12] P. Lin and G.-H. Tu, "An improved GGSN failure restoration mechanism for UMTS," in *ACM Wireless Netw.*, 2005. to be published.

[13] P. Lin, Y.-B. Lin, and I. Chlamtac, "Modeling frame synchronization for UMTS high-speed downlink packet access," *IEEE Trans. Veh. Technol.*, vol. 52, no. 2, pp. 132–141, Apr. 2003.

**Phone Lin** (M'02) received the B.S.C.S.I.E. and Ph.D. degrees 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 Department of Computer Science and Information Engineering (CSIE), National Taiwan University, Taipei, Since August 2004, he has been an Associate Professor in the Department of CSIE, National Taiwan University. His current research interests include personal communications services, wireless Internet, and performance modeling.

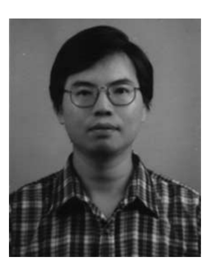
Dr. Lin is a Guest Editor for IEEE WIRELESS COMMUNICATIONS Special Issue on Mobility and Resource Management and a Guest Editor for the ACM/Springer special issue on wireless broad access. He is also an Associate Editorial Member for the *WCMC Journal*.

**Yi-Bing Lin** (M'96–SM'96–F'04) is Chair Professor and Vice President of Research and Development, National Chiao Tung University, Hsinchu, Taiwan, R.O.C. His current research interests include wireless communications and mobile computing. He has published over 190 journal articles and more than 200 conference papers. He is the co-author of the book *Wireless and Mobile Network Architecture* (with I. Chlamtac; New York: Wiley).

Prof. Lin is an ACM Fellow, an AAAS Fellow, and an IEE Fellow.

**Chai-Hien Gan** was born in Malaysia in 1971. He received the B.S. degree in computer science from Tamkang University, Taipei County, Taiwan, R.O.C., in 1994, and the M.S. and Ph.D. degrees in computer science and information engineering from 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 and Information Engineering (CSIE), National Chiao Tung University, Hsinchu, Taiwan, R.O.C. His current research interests include wireless mesh networks, mobile computing, personal communications services, and wireless Internet.



**Jeu-Yih Jeng** received the B.S. degree in mathematics from Fu-Jen University, Taipei, Taiwan, R.O.C., in 1983, the M.S. degree in applied mathematics from National Chiao-Tung University, Hsinchu, Taiwan, in 1985, and the Ph.D. degree in computer science and information engineering from National Chiao-Tung University, Hsinchu, Taiwan, in 1988.

Since 1985, he has been with the Information Technology Laboratory of Telecommunication Laboratories, Chunghwa Telcom Co., Ltd, Taiwan, where he is currently a Distinguished Researcher and a Project Manager. His research interests include design and analysis of personal communications services networks, development of telecommunication operation support systems, and erformance modeling.