Performance Analysis of RLC/MAC and LLC Layers in a GPRS Protocol Stack

K. Premkumar and A. Chockalingam, *Senior Member, IEEE*

Abstract—**In this paper, we analyze the performance of various layers of the general packet radio service (GPRS) protocol stack, including radio link control/medium-access control (RLC/MAC) layer and logical link-control (LLC) layer on the uplink. In the GPRS MAC protocol, several time-slotted uplink radio-frequency channels are shared by the mobiles on a request-reservation-based multiple-access scheme. Using the theory of Markov chains, we derive expressions for the average throughput and delay performance of the GPRS MAC protocol. We evaluate the performance of the RLC layer (in acknowledged mode) using block-level retransmission (BLR), as defined in the current GPRS standard, and compare it with that of using slot-level retransmission (SLR). We show that SLR at the RLC layer performs significantly better than the BLR, particularly when the channel-error rates are moderate to high. We further investigate the choice of parameters (e.g., number of retransmission attempts) for the automatic repeat request schemes at the RLC and LLC layers. Our results show that it is more beneficial to do error recovery by allowing more retransmission attempts at the RLC layer than at the LLC layer. We also evaluate the performance of transmission-control protocol with BLR and SLR at the RLC layer.**

Index Terms—**Automatic repeat request (ARQ), block-level retransmission (BLR), general packet radio service (GPRS), logical link control (LLC), radio link control/medium-access control (RLC/MAC), slot-level retransmission (SLR), transmission-control protocol (TCP).**

I. INTRODUCTION

THE first- and second-generation (1G and 2G) cellular systems, including advanced mobile phone service (AMPS), global system for mobile communications (GSM), IS-136, and IS-95, are circuit-switched (CS) systems designed primarily for voice communications. In addition to voice services, CS data communications at low rates (maximum 14.4 kb/s) are supported by 1G and 2G systems [e.g., cellular digital packet data (CDPD) on AMPS and digital fax and asynchronous data transfer on GSM, IS-95]. With the increasing demand for wireless mobile Internet access at high speeds, the evolution of 2G systems to next-generation systems has been focused on supporting "packet-switched operation over the air" in order to make more efficient use of radio resources. For example, general packet radio services (GPRS) and enhanced data rates

for GSM evolution (EDGE) are systems that upgrade (with packet-routing nodes) and use the existing GSM infrastructure to provide high-speed (maximum 270 kb/s in GPRS and 384 kb/s in EDGE) packet-mode data communications [1]–[3]. The design and performance of protocol stacks in such wireless systems are largely influenced by the erroneous nature of the wireless channel (due to distance losses, shadowing, and multipath fading), as well as the behavior of the commonly used higher layer protocols that are designed primarily for wireline channels that exhibit very low error rates. Transmission-control protocol (TCP), a widely used transport-layer protocol, is one such higher layer protocol [4]. It is well known that TCP performs poorly on wireless channels in which error rates are typically high [5]–[7]. This is because the channel-induced errors often trigger the congestion control mechanism in TCP in such a way that the transmission window size is reduced, thus significantly degrading the TCP throughput. Hence, the lower layers in wireless protocol stacks [e.g., link layer (LL) and media access-control layer (MAC)] must be designed to address these issues of high error rates and higher layer-performance concerns.

Forward error correction (FEC) and automatic repeat request (ARQ) are commonly used error-control techniques on wireless channels. In FEC, redundant bits are added to information bits to detect and correct channel-induced errors. In ARQ, on the other hand, error control is achieved not through error correction, but through the retransmission of erroneous data packets. While FEC is applied at the physical layer, ARQ can be applied at different layers of the protocol stack. Errors uncorrectable by FEC at the physical layer can be handled by the higher layer ARQs. For example, the protocol stack in GPRS consists of ARQs at the radio link-control (RLC) and logical link-control (LLC) layers, in addition to employing FEC at the physical layer [2]. Our focus in this paper is the performance analysis of some of the key layers of the GPRS protocol stack.

Several studies have investigated the performance of the various layers of the GPRS protocol stack, but mainly through simulations [8]–[13]. One of our key contributions in this paper is the analytical approach to performance evaluation of some of the key GPRS protocol layers, including MAC and RLC layers on the uplink. In the GPRS MAC protocol, several time-slotted uplink radio-frequency channels are shared by the mobiles on a request-reservation-based multiple-access scheme. Using the theory of Markov chains, we derive expressions for the average throughput and delay performance of the GPRS MAC protocol [14]. We also evaluate the performance of the RLC layer (in acknowledged mode) using block-level retransmission (BLR), as defined in the current GPRS standard, and compare it with

The authors are with the Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore 560 012, India (e-mail: prem@pal.ece.iisc.ernet.in; achockal@ece.iisc.ernet.in).
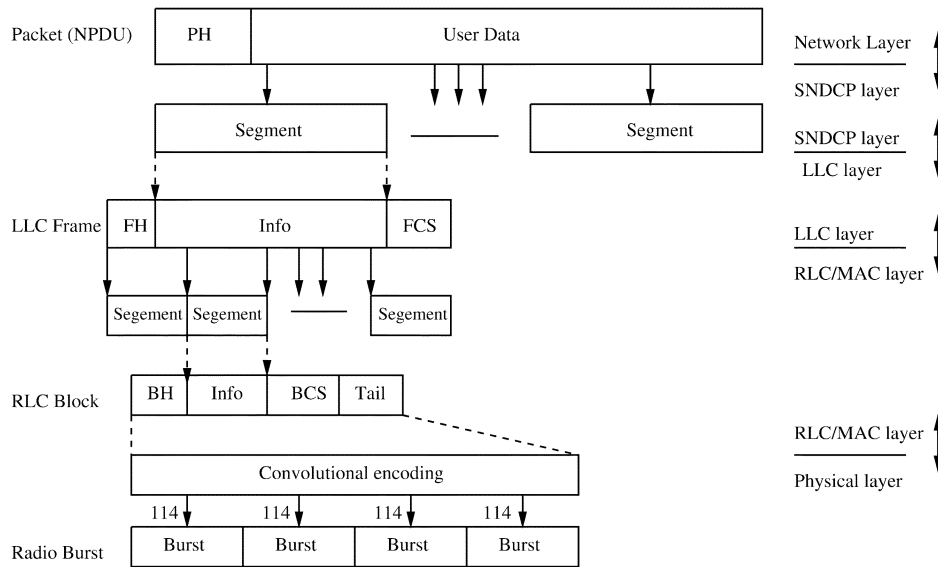
Fig. 1.   Network-layer PDU segmentation into LLC frames, RLC blocks, and MAC bursts.

that of using slot-level retransmission (SLR). We show that SLR at the RLC layer performs better than the BLR, particularly when the channel-error rates are high. We further investigate the choice of parameters (e.g., number of retransmission attempts) for the ARQ schemes at the RLC and the LLC layers. Our results show that it is more beneficial to do error recovery by allowing more retransmissions at the RLC layer than at the LLC layer. We also evaluate the performance of TCP with BLR and SLR at the RLC layer.

The rest of this paper is organized as follows. In Section II, we present the functionalities of the GPRS LLC, RLC, and MAC layers that are relevant to our performance analysis. The performance analyses of the GPRS MAC and RLC layers are presented in Section III. The performance of BLR and SLR at the RLC layer are compared. LLC and TCP performance results with BLR and SLR at the RLC layer are presented in Section IV and the conclusion is presented in Section V.

II. GPRS LLC/RLC/MAC LAYERS

GPRS networks provide Internet protocol (IP) connectivity to mobile users through GSM infrastructure using additional support nodes for packet routing, namely, serving GPRS support node (SGSN) and gateway GPRS support node (GGSN), and the associated protocol stacks. Refer to [1] and [2] for an introduction to GPRS architecture, concepts, system operation, and protocol functions. The over-the-air communication between the mobile station (MS) and the GPRS network is defined by the physical- and data-link-layer functionalities. The physical layer functions include modulation, demodulation, channel coding/decoding, etc. The data-link layer consists of two sublayers, namely, logical link control (LLC) and the radio link control/medium-access control (RLC/MAC). In this section, we present the functionalities of the LLC, RLC, and MAC layers that are relevant to our performance analysis.

LLC Layer: The LLC layer operates between the MS and the SGSN and provides a logical link between them [15]. Packet data units (PDUs) from higher layers (IP layer) are segmented

into variable-size LLC frames (see Fig. 1). The functions of the LLC layer includes link-level flow control and ciphering. The LLC layer can operate either in an *acknowledged* mode or in an *unacknowledged* mode. In the unacknowledged mode of operation, the LLC layer does not attempt the recovery of erroneous LLC frames. LLC frames, erroneously received or otherwise, are passed on to the higher layers. In the acknowledged mode, the LLC layer provides an ARQ mechanism to retransmit erroneous LLC frames. A frame-check sequence (FCS) is provided in each LLC frame to detect LLC frame errors. A retransmission count variable N200 is defined [15]. The LLC is reset and error recovery is passed on to higher layers (e.g., TCP) if LLC frames errors could not be recovered within N200 retransmission attempts.

RLC Layer: The RLC layer is provided below the LLC layer and above the MAC layer [16]. The RLC peers are at the MS and the base station system (BSS). On the transmit side, the RLC layer segments each LLC frame into several RLC data blocks (see Fig. 1). Each RLC data block occupies four time slots, irrespective of the type of channel coding used. Coding schemes CS-1, CS-2, CS-3, and CS-4 are defined with rate-1/2, rate-2/3, rate-3/4, and rate-1 (i.e., no coding), respectively [16]. In the case of coding scheme CS-1, each RLC block consists of 181 information bits, 40 block-check-sequence (BCS) bits, and seven tail/control bits. With single-slot operation [i.e., only one slot per GSM time-division multiple-access (TDMA) frame is allotted to a user], the information rate of coding scheme CS-1 is 9.05 kb/s (i.e., 181 b in four TDMA frames, where each TDMA frame occupies 4.615 ms). Similarly, the maximum information rates possible using other coding schemes are 13.4 kb/s for CS-2, 15.6 kb/s for CS-3, and 21.4 kb/s for CS-4 (see Fig. 2). With multislot operation (i.e., allocation of up to eight slots in a TDMA frame to a user), these maximum possible information rates are increased eight-fold.

Like the LLC, the RLC too can operate either in an *acknowledged* or *unacknowledged* mode. In the acknowledged mode, the RLC provides a selective repeat ARQ mechanism to recover erroneous RLC data blocks. A BCS is provided in each
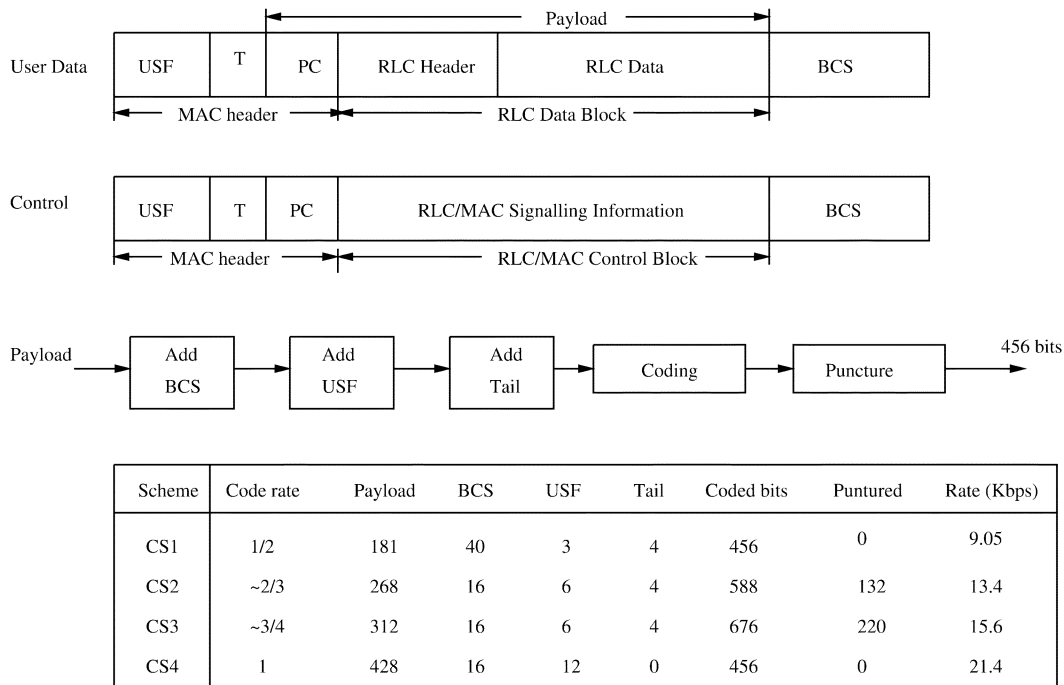
Fig. 2. RLC/MAC block structure and coding.

Scheme	Code rate	Payload	BCS	USF	Tail	Coded bits	Puntured	Rate (Kbps)
CS1	1/2	181	40	3	4	456	0	9.05
CS2	~2/3	268	16	6	4	588	132	13.4
CS3	~3/4	312	16	6	4	676	220	15.6
CS4	1	428	16	12	0	456	0	21.4

RLC data block to enable error detection. A RLC retransmission counter N3104, which keeps track of the number of times a RLC block is retransmitted in the case of error, is defined [16]. The RLC layer is allowed to attempt a maximum of N3104_MAX retransmissions to recover blocks in error. If an erroneous block is not recovered within N3104_MAX retransmission attempts, then control is passed on to LLC to recover the error at the LLC frame level. In the RLC unacknowledged mode, there is no retransmission of erroneous RLC data blocks. On the receive side, the RLC performs the reassembly of LLC frames.

MAC Layer: The GPRS MAC protocol operates on a slotted-ALOHA-based reservation protocol [16]. The MAC layer peers are at the MS and BSS. The MAC layer requests/reserves resources in terms of the number of traffic data slots. The MAC function provides arbitration between multiple mobiles attempting to transmit simultaneously and provides collision detection and recovery procedures.

The packet random-access channel (PRACH) is used by all the mobiles, on a contention basis, for the purpose of sending resource-request packets. Typically, TS0 slot in a GSM frame of eight slots can be used as PRACH. All mobiles are allowed to transmit on PRACH slots, following the slotted-ALOHA protocol. Depending on the system load, the number of PRACHs can be increased. The packet data traffic channels (PDTCH), on the other hand, are used for the transfer of traffic data packets. Resource requests are made by the mobiles in terms of the number of uplink PDTCH slots required. Based on these requests, PDTCH slots are dynamically assigned to the mobiles by the base station (BS). Allocation can be done based on either *single-slot* or *multislot* operation. If only one slot per TDMA frame is assigned to a user, then it is called single-slot operation, whereas if more than one slot (up to eight slots) per TDMA frame is assigned to a user, then it is called multislot operation.

When the MAC at the mobile side receives RLC data blocks to be transferred to the BS, it sends a request packet on the immediately following PRACH slot. The request packet indicates K, the number of PDTCH slots required. If the BS receives the request packet without collision or channel errors and if PDTCH slots are available to honor the request, the BS informs the reservation information to the mobile on the downlink packet-access-grant channel (PAGCH) channel. The reservation information include the PDTCH frequency-time slots that can be used by the mobile for data transfer. The mobile then sends data in those K reserved slots. On the other hand, if the request packet is lost (due to collision or channel errors) or if PDTCH slots are not available, then the mobile will not get the reservation. The mobile will then reschedule its request packet-retransmission attempt to a later time (typically, after a random backoff time).

The MAC control parameters include MAX_RETRANS, PERSISTENCE_LEVEL, as defined in [16]. The MAC layer can send channel requests on the PRACH slots up to a maximum of MAX_RETRANS $+ 1$ retransmission attempts in the event of loss due to collision or channel errors. The delay between retransmission attempts is defined by the PERSISTENCE_LEVEL.

Thus, in summary, in terms of error recovery at different layers: 1) the MAC layer attempts to resolve collision of request packets; 2) RLC layer attempts to recover RLC data block errors through a selective repeat ARQ mechanism; and 3) LLC attempts recovery of erroneous LLC frames through another ARQ mechanism. Link errors unresolved at the LLC layer are passed on to higher layers (e.g., transport layer) to resolve.

III. ANALYSIS OF GPRS MAC PROTOCOL

In this section, we present a Markov-chain-based analysis to evaluate the throughput and delay performance of the GPRS
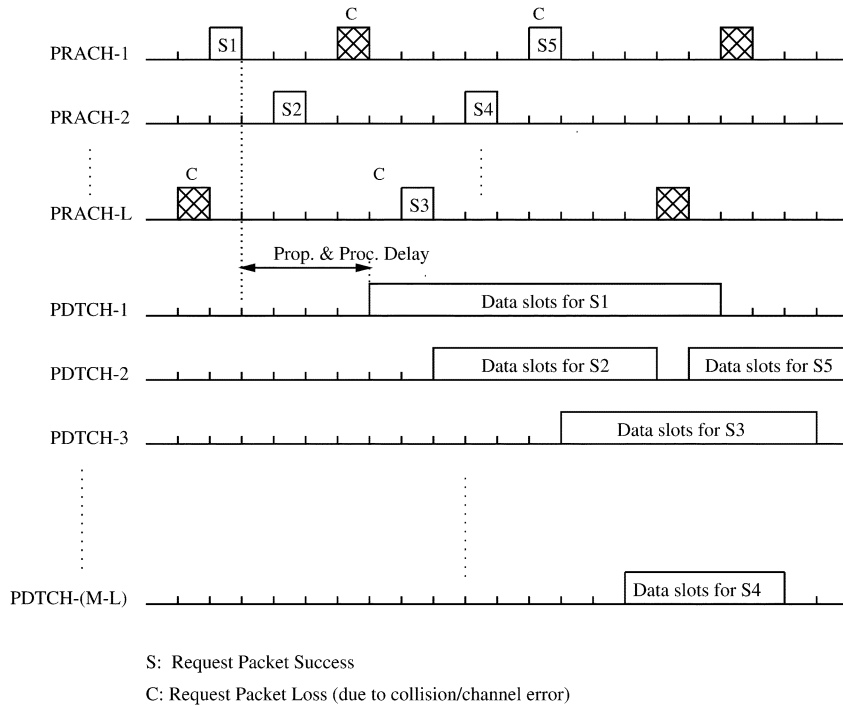
S: Request Packet Success

C: Request Packet Loss (due to collision/channel error)
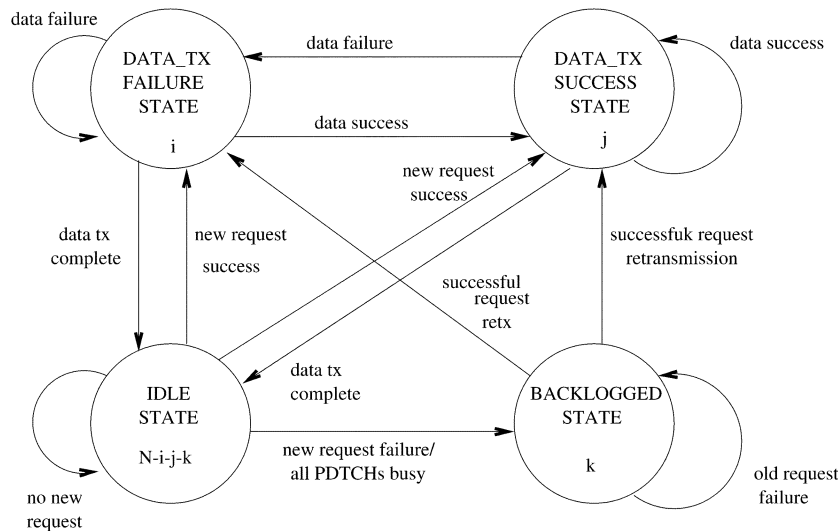
Fig. 3.   GPRS MAC protocol operation.

Fig. 4.   MS state transition diagram.

MAC protocol. Consider a single-cell GPRS system with M, $M \geq 2$ uplink channels and N mobile users. Each channel corresponds to a frequency-time slot pair in the MS-to-BS direction. Out of M channels, L, $1 \leq L < M$, channels are used as PRACHs and the remaining $M - L$ channels are used as PDTCHs. Typically, slot TS0 in all GSM TDMA frames on a given frequency can form a PRACH. Likewise, on a given frequency, slot TS1 in all GSM TDMA frames can form PDTCH-1, slot TS2 can form PDTCH-2, and so on. We consider a single-slot operation in which only one slot per GSM TDMA frame is assigned to a user. For example, TS1 slots in consecutive TDMA frames $n, n + 1, \ldots, n + K$ being assigned to a mobile for data transfer is a typical illustration of the single-slot operation. Considering single-slot operation, all M uplink chan-

nels can be modeled as synchronized slotted channels, as shown in Fig. 3. One request packet is one slot in size. One network (NW) layer PDU, including LLC/RLC headers and checksums, occupies several slots. Between the successful transmission of a request packet on a PRACH slot and the corresponding data transmission on the assigned PDTCH slots, some finite time is elapsed because of the propagation and processing delays involved. We are interested in analyzing the throughput and delay performance of the GPRS MAC protocol. In order to carry out the performance analysis, we assume the following.

1) NW layer PDU arrival process (hence, the new request packet generation process) is Bernoulli with arrival probability λ in each slot. The number of LLC frames per NW layer PDU is assumed to be one. A new NW layer PDU is accepted

only after the completion of the transfer of the previously accepted NW layer PDU.

2) Length of the NW layer PDU (including LLC/RLC headers and checksums), measured in number of slots, is geometrically distributed with parameter g_d, $0 < g_d < 1$.

3) Loss of request packets on PRACH is only due to collision.

4) Retransmission attempts of request packets following a collision on PRACH (or nonavailability of PDTCH) are delayed by geometrically distributed random delays with parameter g_r, $0 < g_r < 1$. This parameter g_r essentially models the backoff delay (in the event of request packet loss), which is characterized by the standards-defined parameter PERSISTENCE_LEVEL.

5) Propagation and processing delays are assumed to be negligible. This assumption can be valid in our considered system of single-slot operation in which the response from the BS can come within one TDMA frame time itself.

As per the GPRS MAC protocol and the system model described above, the mobile can be in any one of the following states in a slot, namely, the *idle*, *backlogged*, *data_Tx_success*, or *data_Tx_failure* states, as shown in Fig. 4. In the idle state, a mobile remains idle with probability $(1 - \lambda)$ and generates a PDU with probability λ. If n different mobiles send their request packets on the same PRACH slot, then one or more request packets can be received correctly, depending on the capture probability. The mobiles who lost their request packets go to the backlogged state. If a mobile's request packet is successfully received, but there are no available PDTCHs to serve the request, then also the mobile goes to the backlogged state. On the other hand, if there are available PDTCHs to serve a successfully received request, then the mobile goes to the data_Tx states, where it sends data on the assigned PDTCH slots. The data_Tx states include the data_Tx_success and data_Tx_failure states, corresponding to the transmission on a PDTCH slot being, respectively, a success or failure.

A. Throughput Analysis

Let $\{\mathbf{D}_{ft}; t \in \{1, 2, 3, \ldots\}\}$ represent the number of mobiles in the data_Tx_failure state, $\{\mathbf{D}_{st}; t \in \{1, 2, 3, \ldots\}\}$ be the number of mobiles in the data_Tx_success state, and $\{\mathbf{B}_t; t \in \{1, 2, 3, \ldots\}\}$ be the number of mobiles in the backlogged state, at the beginning of slot t. The three-dimensional (3-D) process $\{\mathbf{Z}_t = (\mathbf{D}_{ft}, \mathbf{D}_{st}, \mathbf{B}_t); t \in \{1, 2, 3, \ldots\}\}$ defines the state of the system at time t. Now, let us analyze the transition of the system from $\mathbf{z_1} = (i_1, j_1, k_1)$ in slot t to $\mathbf{z_2} = (i_2, j_2, k_2)$ in slot $t + 1$, $0 \leq i_1 \leq M - L$, $0 \leq j_1 \leq M - L - i_1$, $0 \leq k_1 \leq N - i_1 - j_1$, $0 \leq i_2 \leq M - L$, $0 \leq j_2 \leq M - L - i_2$, $0 \leq k_2 \leq N - i_2 - j_2$. Refer to Fig. 5, which illustrates the state transitions from slot t to slot $t + 1$.

During slot t, b_i mobiles $(0 \leq b_i \leq i_1)$ out of i_1 mobiles in the data_Tx_failure state have data to transmit and the remaining $(i_1 - b_i)$ mobiles end the data transmission and go to the idle state. Out of those b_i mobiles, f_s mobiles, $0 \leq f_s \leq b_i$, go to the data_Tx_success state and the remaining $(b_i - f_s)$ mobiles stay in the data_Tx_failure state.
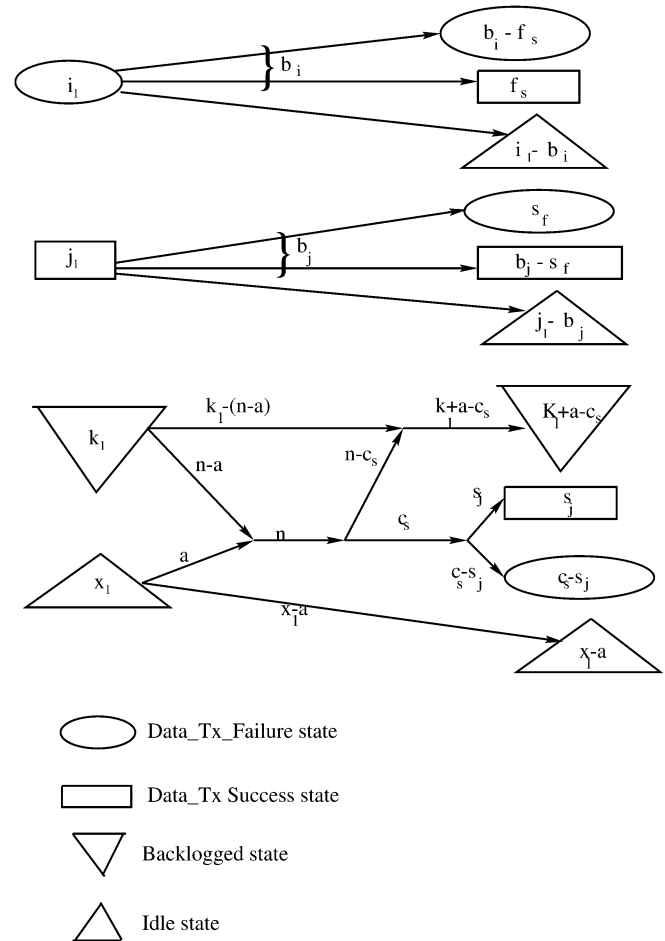

Fig. 5. State transitions of a mobile from slot t to slot $t + 1$.

Also, b_j mobiles $(0 \leq b_j \leq j_1)$ out of j_1 mobiles in the data_Tx_success state have data to transmit and the remaining $(j_1 - b_j)$ mobiles end data transmission and go to the idle state. Out of those b_j mobiles, s_f mobiles $0 \leq s_f \leq b_j$ go to the data_Tx_failure state and the remaining $(b_j - s_f)$ mobiles stay in the data_Tx_success state.

Out of k_1 mobiles in the backlogged state, $(n - a)$ mobiles attempt request packet transmission and the remaining $(k_1 - (n - a))$ mobiles stay in the backlogged state. Also, a idle mobiles send request packets giving a total of n requests, out of which c_s, $0 \leq c_s \leq n$ succeeds. The remaining $(n - c_s)$ mobiles go to the backlogged state. Out of c_s succeeding mobiles, s_j mobiles go to the data_Tx_success state and the remaining $(c_s - s_j)$ go to the data_Tx_failure state.

So, at the end of slot t (or at the beginning of slot $t + 1$), the total number of mobiles in the data_Tx_failure state is $(b_i - f_s + s_f + c_s - s_j)$, in the data_Tx_success state it is $(f_s + b_j - s_f + s_j)$, and in the backlogged state is $(k_1 + a - c_s)$. Assuming that slot errors can occur independently from slot to slot and denoting P_s to be the probability of slot error,[1] transition probability $\tilde{P}_{z_1, z_2}$

---

[1]It is noted that the slot-error rate P_s depends on the channel signal-to-noise ratio (SNR) and the modulation and coding used. The modulation and coding schemes can change dynamically in a practical GPRS system depending on the channel conditions. The mapping of the channel SNR to the slot-error rate is beyond the scope of this paper. References [10] and [13] give such mapping between C/I to block error rate for various coding schemes.

can be written as shown in (1) at the bottom of the page where $a = k_2 - k_1 + c_s$, n is the number of requests sent, c_s is the number of successful requests in slot t, $b_j = j_2 - s_j - f_s + s_f$, $b_i = i_2 - c_s + s_j - s_f + f_s$, $f_s \le b_i$, $s_f \le b_j$, and $f(c_s, n, L) =$ Prob (c_s successes given that n requests are sent and that L PRACHs are available), which can be written as

$$f(c_s, n, L) = \sum_{l=0}^{n} \sum_{c=0}^{1} \binom{n}{l} \left(\frac{1}{L} \right)^l \left(1 - \frac{1}{L} \right)^{n-l}$$
$$\cdot \text{cap}(l, c) \cdot f(c_s - c, n - l, L - 1) \quad (2)$$

where $\text{cap}(l, c)$ is the probability of capturing c out of l colliding request packets. Equation (2) is terminated by fixing $f(c, n, 1) = \text{cap}(n, c)$. It is noted that the capture probability in the above can be obtained either by its analytical evaluation for a given channel model (for example, as derived in [17] and [18] for Rayleigh- and Rician-fading channels with log-normal shadowing) or by using empirical capture models (for example, as given by the European Telecommunications Standards Institute (ETSI) in [19]), which has been used in several papers (e.g., [8]). In this paper, we use the empirical capture model given in the ETSI standard [19]. As in [19], we set $\text{cap}(0,1) = 0$, $\text{cap}(1,1) = 1$, $\text{cap}(2,1) = 0.67$, $\text{cap}(3,1) = 0.48$, $\text{cap}(4,1) = 0.40$, $\text{cap}(5,1) = 0.35$, $\text{cap}(n,1) = 0$ for $n > 5$, and $\text{cap}(n,0) = 1 - \text{cap}(n,1)$.

The above analysis assumes a large number of PDTCHs, so that a successful request always gets an assignment. If the number of PDTCHs is small compared to the number of users N, (i.e., $M - L < N$), then a successful request may not get an assignment, as all the PDTCHs may be busy. This event occurs if $M - L - (L - 1) \le (i_1 + j_1) \le M - L$ and $M - L - (L - 1) \le (i_2 + j_2) \le (i_1 + j_1)$. This implies that the total number of idle channels $M - L - (i_1 + j_1) < L$. If $l \le L$ requests succeed in any time slot and the number of idle

channels $M - L - (i_1 + j_1)$ is less than l, then all l requests are backlogged. For those z_1 and z_2 satisfying the above condition, compute the probability $\hat{P}_{z_1, z_2}$ and add to the corresponding $\tilde{P}_{z_1, z_2}$ term calculated using (1). To do that, let x represent the number of idle channels that is given by $M - L - (i_1 + j_1)$. y represents the number of data transmissions ended during slot t, which is given by $(i_1 + j_1) - (i_2 + j_2)$, and $\mu = k_2 - k_1$ represents the new arrivals that are backlogged due to nonavailability of channels. The probability $\hat{P}_{z_1, z_2}$ for the transition from $\mathbf{z_1} = (i_1, j_1, k_1)$ to $\mathbf{z_2} = (i_2, j_2, k_2)$ is then given by

$$\hat{P}_{z_1, z_2} = \sum_{\theta=0}^{k_1} \sum_{c_s=y+x+1}^{L} \sum_{s_f=0}^{i_2} \sum_{f_s=0}^{j_2} \binom{N - i_1 - j_1 - k_1}{\mu}$$
$$\cdot \lambda^{\mu} (1 - \lambda)^{N - i_1 - j_1 - k_1 - \mu}$$
$$\cdot \binom{k_1}{\theta} g_r^{\theta} (1 - g_r)^{k_1 - \theta} \cdot \binom{i_1}{b_i} g_d^{i_1 - b_i} (1 - g_d)^{b_i}$$
$$\cdot \binom{b_i}{f_s} (1 - P_s)^{f_s} P_s^{b_i - f_s}$$
$$\cdot \binom{j_1}{b_j} g_d^{j_1 - b_j} (1 - g_d)^{b_j} \cdot f(c_s, \mu + \theta, L)$$
$$\cdot \binom{b_j}{s_f} P_s^{s_f} (1 - P_s)^{b_j - s_f} \quad (3)$$

where $b_j = j_2 - f_s + s_f$, $b_i = i_2 - s_f + f_s$, $f_s \le b_i$, $s_f \le b_j$. Thus

$$P_{z_1, z_2} = \tilde{P}_{z_1, z_2} + \hat{P}_{z_1, z_2}. \quad (4)$$

The Markov chain $\{\mathbf{Z}_t; t \in \{1, 2, 3, \ldots\}\}$ has a finite number of states and is positive recurrent [20]. Hence, it has a stationary steady-state distribution and is found by solving

$$\Pi = \Pi \mathbf{P} \quad (5)$$

$$\tilde{P}_{z_1, z_2} = \Pr\big(\mathbf{z_2} = (i_2, j_2, k_2) | \mathbf{z_1} = (i_1, j_1, k_1)\big)$$

$$= \sum_{n=0}^{N - i_1 - j_1} \sum_{c_s=0}^{L} \sum_{s_j=0}^{c_s} \sum_{s_f=0}^{j_1} \sum_{f_s=0}^{i_1} \Pr\Big\{ n, c_s, s_j, s_f, f_s, \mathbf{z_2} = (i_2, j_2, k_2) | \mathbf{z_1} = (i_1, j_1, k_1) \Big\}$$

$$= \sum_{n=0}^{N - i_1 - j_1} \sum_{c_s=0}^{L} \sum_{s_j=0}^{c_s} \sum_{s_f=0}^{j_1} \sum_{f_s=0}^{i_1} \Pr\{\text{`}a\text{' new arrivals}\}$$

$\cdot \Pr\{\text{request packet transmissions from } (n - a) \text{ backlogged mobiles}\}$

$\cdot \Pr\{b_i \text{ out of } i_1 \text{ continue to tx data}\} \cdot \Pr\{f_s \text{ out of } b_i \text{ go to success}\}$

$\cdot \Pr\{b_j \text{ out of } j_1 \text{ continue to tx data}\} \cdot \Pr\{s_f \text{ out of } b_j \text{ go to failure}\}$

$\cdot \Pr\{c_s \text{ out of } n \text{ requests succeed and } s_j \text{ of } c_s \text{ go to success}\}$

$$= \sum_{n=0}^{N - i_1 - j_1} \sum_{c_s=0}^{L} \sum_{s_j=0}^{c_s} \sum_{s_f=0}^{j_1} \sum_{f_s=0}^{i_1} \binom{N - i_1 - j_1 - k_1}{a} \lambda^a (1 - \lambda)^{N - i_1 - j_1 - k_1 - a}$$

$$\cdot \binom{k_1}{n - a} g_r^{n-a} (1 - g_r)^{k_1 - n + a} \cdot \binom{i_1}{b_i} g_d^{i_1 - b_i} (1 - g_d)^{b_i} \cdot \binom{b_i}{f_s} (1 - P_s)^{f_s} P_s^{b_i - f_s}$$

$$\cdot \binom{j_1}{b_j} g_d^{j_1 - b_j} (1 - g_d)^{b_j} \cdot \binom{b_j}{s_f} P_s^{s_f} (1 - P_s)^{b_j - s_f} \cdot f(c_s, n, L) \cdot \binom{c_s}{s_j} (1 - P_s)^{s_j} P_s^{c_s - s_j} \quad (1)$$

where $\Pi = [\pi_{ijk}]$, $0 \le i \le M - L$, $0 \le j \le M - L - i$, $0 \le k \le N - i - j$ is the steady-state probability vector. The average system throughput η is obtained as

$$\eta = \sum_{i=0}^{M-L} \sum_{j=0}^{M-L-i} \sum_{k=0}^{N-i-j} j\pi_{ijk}. \tag{6}$$

The average per-channel throughput η_c is then given by

$$\eta_c = \frac{\eta}{M}. \tag{7}$$

B. Delay Analysis

Next, we derive the mean PDU transfer delay performance. The mean PDU transfer delay $\overline{D}$ is defined as the average number of slots elapsed from the slot where a PDU arrived to the slot where the PDU transmission is complete. The number of users in the nonidle state (i.e., data_Tx_failure, data_Tx_success, and backlogged states) contribute to the mean delay. There are $\nu = (i + j + k)$ nonidle users in the system and averaging it over steady-state distribution gives us

$$E(\nu) = \sum_{i=0}^{M-L} \sum_{j=0}^{M-L-i} \sum_{k=0}^{N-i-j} (i + j + k)\pi_{ijk}. \tag{8}$$

There are $N - i - j - k$ idle users and each will generate requests with probability λ in each slot. The average arrival rate to the system is given by

$$\Lambda = \lambda(N - E(\nu)). \tag{9}$$

From Little's theorem, the average time an user spends in the system is given by the ratio between the number of users in the system to the average arrival rate. Hence

$$\overline{D} = 1 + \frac{E(\nu)}{\Lambda}. \tag{10}$$

Note that the one in (10) is added to ensure that there is one slot delay for the mobiles to enter into the nonidle state.

C. RLC Layer—Acknowledged Mode

Note that the analysis in Sections III-A and B corresponds to the RLC/MAC protocol operation with RLC in the unacknowledged mode (i.e., there is no ARQ at the RLC). In the acknowledged mode, however, RLC retransmits erroneous data blocks using a selective repeat ARQ mechanism. Each RLC data block consists of four slots. In our RLC layer analysis here, we consider a SLR mechanism by which a slot in error is repeatedly retransmitted until it succeeds. This implies that the number of retransmission attempts at the RLC layer is infinity. We derive the throughput-delay performance of this SLR scheme as follows.

Let the random variable T represent the length of PDU in number of slots and the random variable Y_i represent the number of transmission attempts of the ith slot until success. Thus, X,

the total number of slots required to successfully transmit a PDU of length T slots, is given by

$$X = \sum_{i=1}^{T} Y_i. \tag{11}$$

The distribution of T is geometric (by assumption) and is given by

$$\Pr(T = t) = g_d(1 - g_d)^{t-1}, \qquad t = 1, 2, 3, \dots \tag{12}$$

and the distribution of Y_i [Y_i's being independent and identically distributed (i.i.d.), since we consider i.i.d. slot errors] is given by

$$\Pr(Y_i = y) = (1 - P_s)(P_s)^{y-1}, \qquad y = 1, 2, 3, \dots \tag{13}$$

where P_s is the slot-error rate. The distribution of X in (11) can be evaluated using Transform techniques, as follows:

$$\begin{aligned}
\phi(z) &= E[z^X] \\
&= E\left[z^{\sum_{i=1}^{T} Y_i}\right] \\
&= E\left[\prod_{i=1}^{T} z^{Y_i}\right] \\
&= E_T\left[E_{Y_i}\left(\prod_{i=1}^{t} z^{Y_i} | T = t\right)\right] \quad (\because T \amalg Y_i) \\
&= E_T\left[\prod_{i=1}^{t} E_{Y_i}\left[z^{Y_i} | T = t\right]\right] \quad (\because Y_i \amalg Y_j, \forall i \ne j) \\
&= E_T\left[\left(\frac{z(1 - P_s)}{1 - zP_s}\right)^T\right] \\
&= \sum_{t=1}^{\infty} \left(\frac{z(1 - P_s)}{1 - zP_s}\right)^t \cdot g_d(1 - g_d)^{t-1} \\
&= \frac{zg_d(1 - P_s)}{1 - z(1 - g_d(1 - P_s))} \\
&= g_d(1 - P_s) \cdot \sum_{j=1}^{\infty} (1 - g_d(1 - P_s))^{j-1} z^j. \tag{14}
\end{aligned}$$

Therefore

$$\begin{aligned}
\Pr(X = x) &= g_d(1 - P_s)[1 - g_d(1 - P_s)]^{x-1} \\
&\quad x = 1, 2, 3, \dots. \tag{15}
\end{aligned}$$

Now, in order to obtain the average throughput and mean delay for the RLC (acknowledged mode) with SLR, we need to simply change the parameter g_d to $g_d(1 - P_s)$ in (1) and (3).

Note that the above analysis is for infinite number of retransmission attempts at the RLC with SLR. The effect of finite number of retransmission attempts at the RLC with SLR is evaluated through simulations. Likewise, the performance using BLR at RLC is also evaluated through simulations.
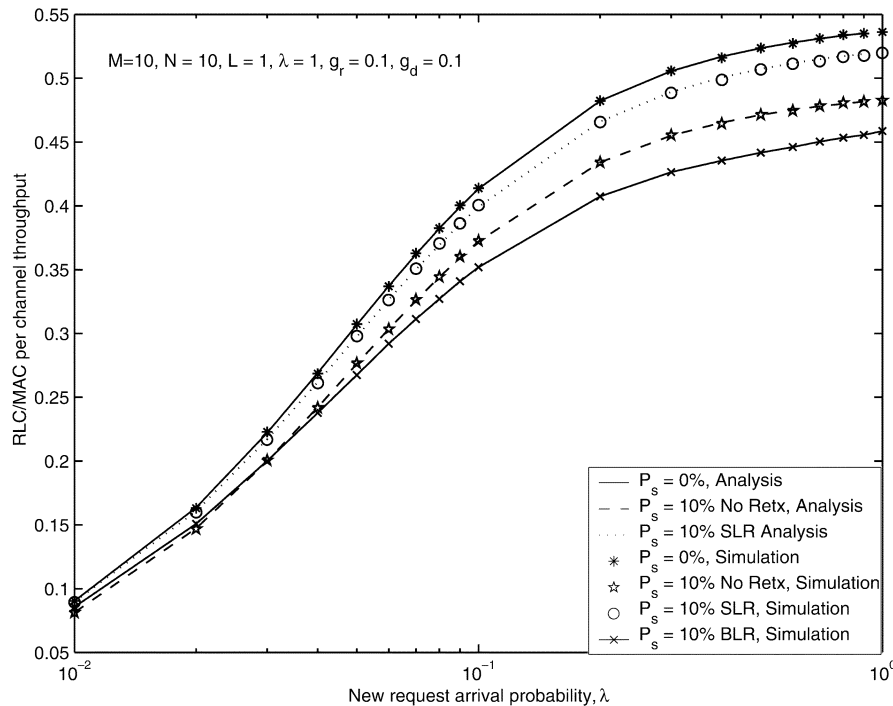
Fig. 6.   RLC/MAC average per channel throughput η_c versus new request arrival probability λ. $N = 10$, $M = 10$, $L = 1$, $g_r = 0.1$, and $g_d = 0.1$.

D. RLC/MAC Performance Results

In this section, we present the results and discussions of the GPRS RLC/MAC performance obtained through both of the analyses above, as well as simulations. In Fig. 6, numerical results for the average per-channel throughput of the RLC/MAC protocol, obtained from (7), for $N = 10$, $M = 10$, $L = 1$, $g_r = 0.1$, and $g_d = 0.1$ are plotted as a function of the new request arrival probability λ. It is noted that the g_r value of 0.1 implies that the average backoff delay after collision is ten slots. Likewise, the g_d value of 0.1 implies that the average length of the PDU measured in number of slots is 10. According to the throughput computation in (7), the PRACH slots do not contribute to the effective throughput. In other words, for M channels out of which L channels are PRACH, the maximum capacity is given by $(M - L)/M$. The effect of slot errors with/without RLC SLR is also plotted for a slot-error rate of 10% (corresponding to poor channel conditions). These results are compared with the corresponding RLC BLR performance obtained through simulation.

It is noted that the average per-channel throughput in Fig. 6 gives the average number of successful slots and the average number of successful blocks for slot level and block retransmissions, respectively. In order to convert this normalized throughput to an equivalent data rate in kilobits/second, the number of information and control bits [including BCS and slot-check sequence (SCS) bits] defined in the various coding schemes CS-1, CS-2, CS-3, and CS-4 needs to be taken into account. For example, when CS-1 is used, a normalized per-channel throughput of, say, 0.45 corresponds to an effective data rate of 4.07 kb/s (i.e., 0.45×9.05 kb/s) for block-level retransmission. For SLR, assuming 10 b of SCS in each slot,

TABLE I
EFFECTIVE PER-CHANNEL THROUGHPUT (IN KILOBITS PER SECOND) FOR BLR VERSUS SLR FOR VARIOUS CODING SCHEMES. $N = 10$, $M = 10$, $L = 1$, $g_r = g_d = 0.1$, $\lambda = 1$, AND $P_s = 10\%$. (SEE FIG. 6)

Coding scheme	FEC	BLR (Kbps)	SLR (Kbps)
CS-1	1/2	4.16	4.52
CS-2	2/3	6.16	6.97
CS-3	3/4	7.17	8.06
CS-4	1(No FEC)	9.84	11.02

the effective data rate becomes 3.9 kb/s (i.e., 0.45×8.7 kb/s). Note that the maximum information rate of 8.7 kb/s in the above is obtained as follows: Each slot carries 114 (rate 1/2) coded bits, so that 40 information bits, ten SCS bits, and seven control bits constitute one slot before coding. Forty information bits sent every 4.615 ms results in an effective information rate of 8.7 kb/s. The effective data rates for other coding schemes CS-2, CS-3, and CS-4 can be obtained likewise. It is noted that when the slot-error rate is zero (or very low values of P_s) BLR performs slightly better than SLR (i.e., maximum information rate of 9.05 versus 8.7 kb/s), which is due to the increased overheads due to the tail/control bits in SLR. However, as we will see subsequently, SLR performs much better than BLR when channel conditions are poor (moderate to high values of P_s).

From Fig. 6 we observe that the throughput results obtained from (7) closely match the results obtained through simulation (in fact, since there are no approximations involved in the analysis, this close match is expected). It is observed that for low values of new request arrival probability λ the system spends most time in the idle state, resulting in very low throughputs. For the case of $\lambda \to 0$, the average time duration between arrivals is $1/\lambda$ and during this time interval the average number of

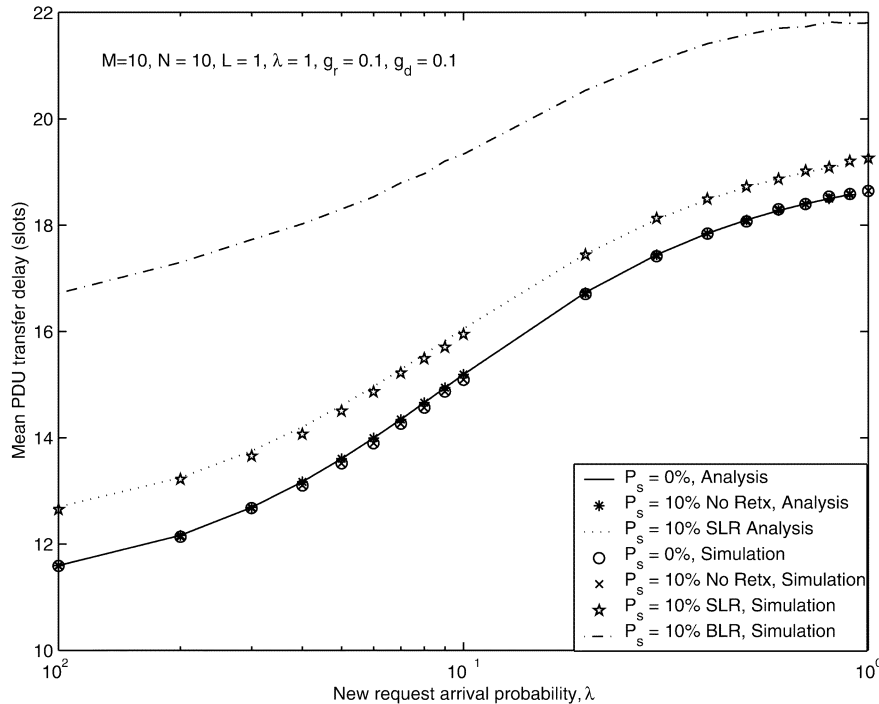Fig. 7. Mean PDU transfer delay (in number of slots) at RLC/MAC versus new request arrival probability λ. $N = 10, M = 10, L = 1, g_r = 0.1$, and $g_d = 0.1$.

successful slots are given by $1/g_d$. Thus, the average per channel throughput is given by

$$\eta_c = (1 - P_s) \cdot \frac{1}{g_d} \cdot \frac{M - L}{M}.$$

For $\lambda = 0.01$, $g_d = 0.1$, $M = 10$, $L = 1$, and $P_s = 0$, η_c is 0.09, which is very close to the value shown in Fig. 6. As λ increases, the fraction of time the system spends in the idle state decreases, which results in increased throughput. When there are no slot errors (i.e., $P_s = 0\%$), all the slots carrying data traffic are successful, which represents the best possible performance. For example, for $\lambda = 0.01$ the per-channel throughput achieved is 0.09 (i.e., an effective data rate of 666 b/s using coding scheme CS-1) and for λ values closer to unity the per-channel throughput increases to 0.54 (4 kb/s using CS-1).

It is noted that the "No Retx" plot in Fig. 6 corresponds to GPRS MAC with RLC in the unacknowledged mode of operation. For high slot-error rates (say, $P_s = 10\%$), when there are no RLC retransmissions, the fraction of successful slots decreases and, hence, the throughput decreases. On the other hand, for the same slot-error rate (of 10%), SLR at the RLC improves the throughput performance. This is because the fraction of time the channel is left idle is reduced due to the retransmission attempts. As long as the slot-error rate is reasonably good, this would result in increased throughput. Another observation in Fig. 6 is that the throughput achieved with BLR at the RLC is much lower than the SLR. This is because, in BLR, even if one slot in a block is in error, the entire block (of four slots) will be retransmitted; this considerably reduces the throughput. For example, for high arrival rates ($\lambda = 1$), the normalized per-channel throughput η_c, achieved using BLR, is 0.46 (effective data rate of 4.16 kb/s using CS-1), whereas SLR achieves a per-channel throughput of 0.52 (effective data rate of 4.52

kb/s using CS-1). Likewise, for these SLR and BLR throughput values at $\lambda = 1$, the effective data rate (in kilobits per second) achieved using CS-2, CS-3, and CS-4 are computed and given in Table I, from which it is observed that gains in throughput are possible with SLR as compared to BLR.

The mean PDU transfer delay performance of the GPRS RLC/MAC protocol is evaluated using (10) for the same set of parameters used in Fig. 6. The mean PDU transfer delay in number of slots is plotted in Fig. 7 as a function of λ. In the case of No Retx (i.e., the RLC unacknowledged mode), it takes the same number of slots to carry the traffic as in the no-error case. Hence, the delay is the same for both the no-error case as well as the error case with no retransmission. The delay for SLR increases as it takes more slots to successfully deliver the data slots. In BLR, since the entire block gets retransmitted even if one slot in a block is in error, the delay performance is worse than SLR. For example, for λ values near unity the mean delay is 19 slots for SLR and 23 slots for BLR.

The effect of the number of channels M on the throughput characteristics of the RLC/MAC protocol is shown in Fig. 8 for $N = 10, L = 1, g_r = 0.1, g_d = 0.1$, and $\lambda = 1$. The total number of channels M is varied in the range from two to ten. From Fig. 8, we observe the following. The per-channel throughput increases as M increases, up to a certain a value of M, beyond which the throughput decreases. This is because, at low values of M, requests are backlogged due to the non-availability of PDTCHs, whereas PDTCHs are idle most of the time at high values of M. More interestingly, around the optimum value of M for the chosen set of parameters and traffic load, SLR significantly outperforms BLR. For example, when $M = 5$, BLR gives a per-channel throughput of 0.47, whereas SLR gives 0.64. The mean PDU transfer delay performance for the same set of parameters used in Fig. 8 is illustrated in Fig. 9.
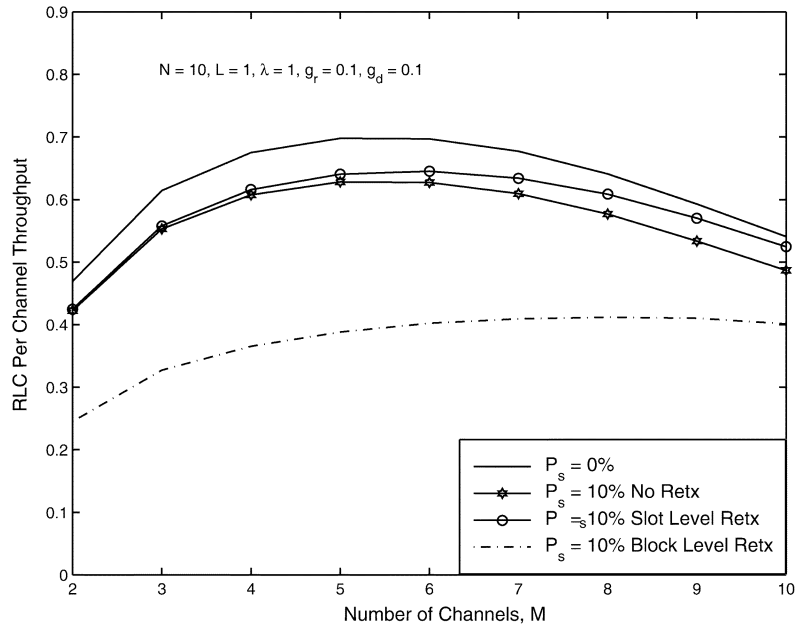
Fig. 8.   RLC/MAC average per-channel throughput η_c versus number of channels M. $N = 10$, $L = 1$, $\lambda = 1$, $g_r = 0.1$, and $g_d = 0.1$.
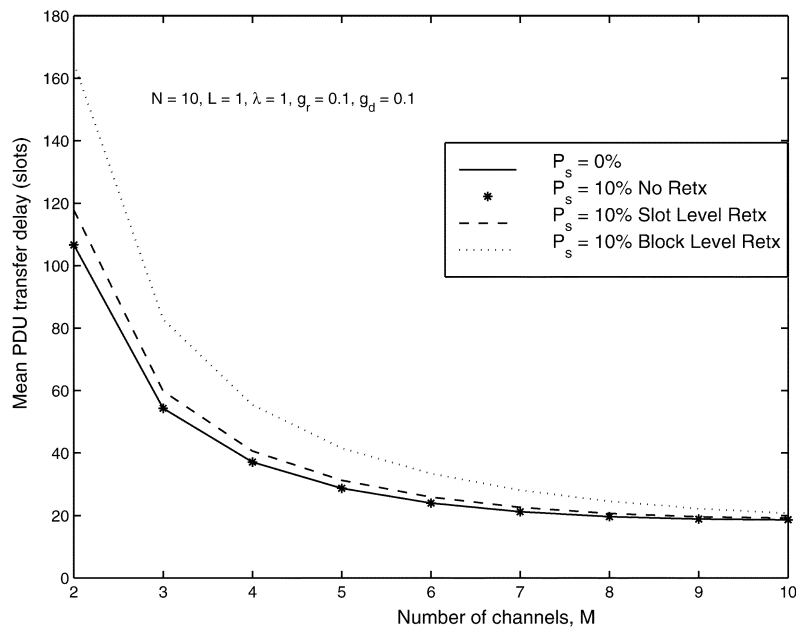

Fig. 9.   Mean PDU transfer delay at RLC/MAC versus number of channels M. $N = 10$, $L = 1$, $\lambda = 1$, $g_r = 0.1$, and $g_d = 0.1$.

The effect of slot-error rate P_s on the per-channel throughput and delay performance is shown in Figs. 10 and 11, for $N = 10$, $L = 1$, $g_r = 0.1$, $g_d = 0.1$, and $\lambda = 1$. The plot shows the performance of RLC in unacknowledged mode (No Retx) as well as the RLC in acknowledged mode with SLR and BLR. As expected, the per-channel throughput decreases as the slot-error rate increases. As mentioned before, when the slot-error rate is small (good channel condition), BLR performs slightly better than SLR due to a larger number of overhead bits in SLR as compared to BLR. However, as the slot-error rate increases, BLR performs more poorly than even the unacknowledged mode. This is because even if one slot goes into error, all the four slots of a block are retransmitted. Also, SLR is found to significantly outperform both unacknowledged mode and BLR. This performance gain is more when the slot-error rate is high. Thus, from

Fig. 6–9, we observe that significant improvement in throughput and delay performance is possible using SLR instead of the BLR defined in the GPRS standard.

IV. LLC AND TCP PERFORMANCE

In this section, we evaluate the throughput of the LLC and TCP, with the RLC/MAC layers below, through simulations. We consider the LLC acknowledged mode of operation. Of particular interest here from the performance point of view is the effect and optimum choice of the maximum LLC retransmission count R_{llc} (LLC parameter N200 defined in the standard) and the maximum RLC retransmission count R_{rlc} (RLC parameter N3104_MAX defined in the standard). In the performance
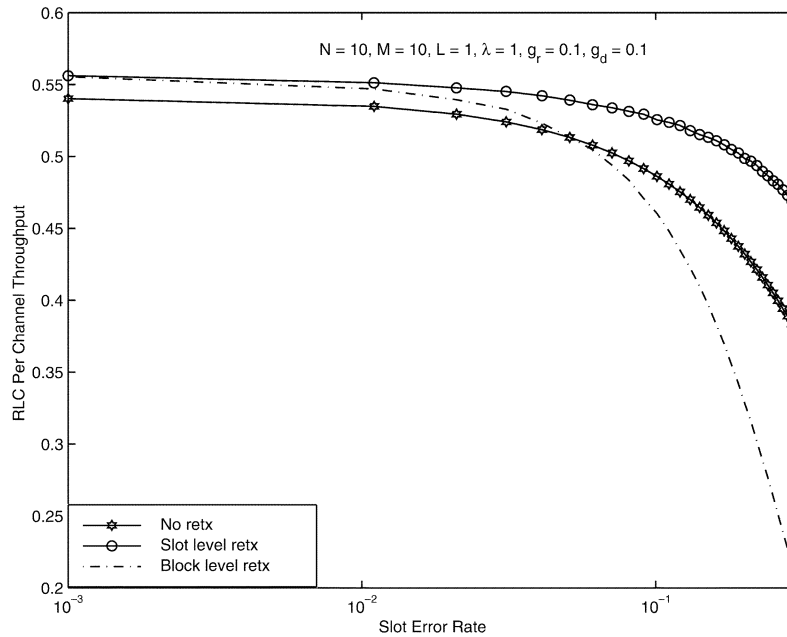
Fig. 10. RLC/MAC average per channel throughput η_c versus slot-error rate P_s. $N = 10$, $M = 10$, $L = 1$, $\lambda = 1$, $g_r = 0.1$, and $g_d = 0.1$.
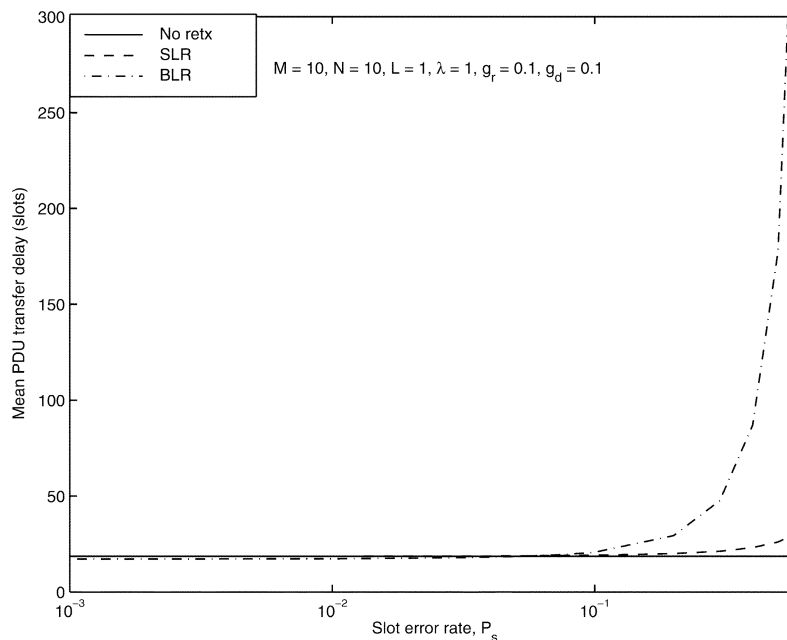

Fig. 11. Mean PDU transfer delay at RLC/MAC versus slot-error rate P_s. $N = 10$, $M = 10$, $L = 1$, $\lambda = 1$, $g_r = 0.1$, and $g_d = 0.1$.

evaluation of the LLC, we use a similar system model as described in Section III, except that here we assume that the message length measured in the *number of LLC frames* is geometrically distributed with parameter g_f. We consider selective acknowledgment of LLC frames [15]. Also, we assume that each LLC frame consists of five RLC blocks (20 slots). If an erroneous LLC frame is not recovered within R_{llc} retransmission attempts, the LLC is reset and reestablished. We assume that this reset and reestablishment delay is RESET_DELAY in the number of slots.

A. LLC Performance Results

The average per-channel throughput at the LLC layer as a function of maximum RLC retransmission count R_{rlc} is plotted

in Fig. 12, for $N = 10$, $M = 10$, $L = 1$, $\lambda = 0.1$, $g_r = 0.1$, $g_f = 0.2$, and RESET_DELAY $= 200$ slots. Both BLR as well as SLR at a slot-error rate of $P_s = 10\%$ are considered. The R_{llc} values considered are 1, 2, and 3. We define the throughput at the LLC layer as the average number of successful LLC frames. From Fig. 12, we observe that, for a fixed R_{llc}, increasing R_{rlc} increases throughput. By increasing R_{rlc}, we try to recover erroneous blocks in the RLC layer itself rather than giving up the entire frame (which contains the erroneous blocks) to the LLC. We also observe that, as R_{llc} is increased, throughput increases. When R_{llc} is small, LLC *resets* occur frequently. A reset makes the channel to stay idle for RESET_DELAY slots. Hence, the per-channel throughput at the LLC is less for small values of R_{llc}. By in-
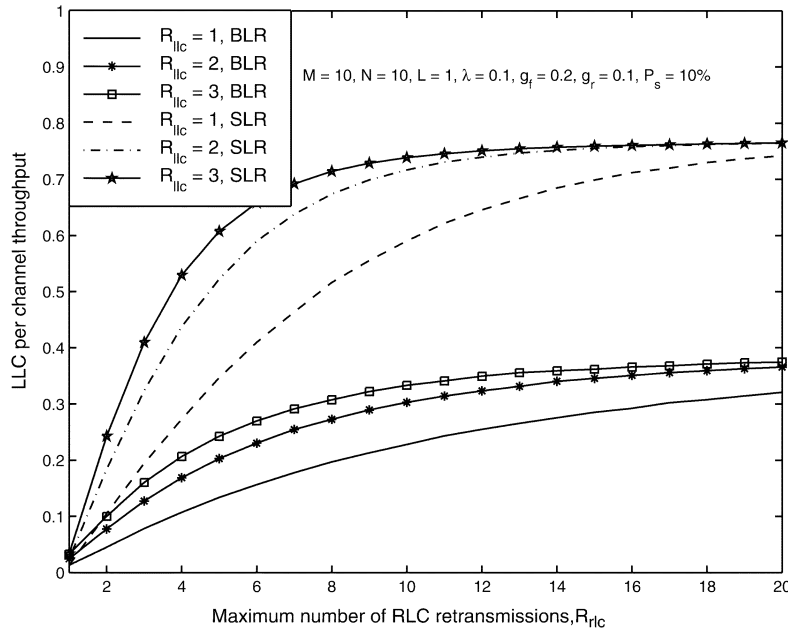
Fig. 12.   LLC average per-channel throughput versus maximum number of RLC retransmissions R_{rlc}. $N = 10$, $M = 10$, $L = 1$, $\lambda = 0.1$, $P_s = 10\%$, $g_r = 0.1$, and $g_f = 0.2$.
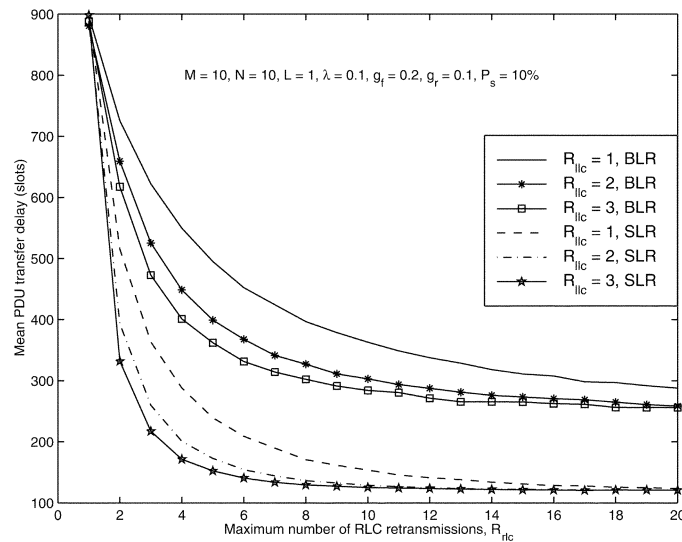

Fig. 13.   Mean PDU transfer delay at LLC versus the maximum number of RLC retransmissions R_{rlc}. $N = 10$, $M = 10$, $L = 1$, $\lambda = 0.1$, $P_s = 10\%$, $g_r = 0.1$, and $g_f = 0.2$.

creasing R_{llc}, we try to avoid too many resets, which results in a better throughput.

The delay performance at the LLC layer for the same system parameter values in Fig. 12 is plotted in Fig. 13. When both R_{rlc} and R_{llc} are small, the LLC resets will occur frequently, which will increase the PDU transfer delay. Thus, for small R_{rlc}, the delay performance improves by increasing R_{llc}. When R_{rlc} is large enough (e.g., $R_{\mathrm{rlc}} > 10$), the delay performance for $R_{\mathrm{llc}} = 2$ or 3 is approximately the same. This is because most of the erroneous RLC blocks are recovered at the RLC layer itself and fewer errors are being passed to the LLC to recover. From Figs. 12 and 13, it can be seen that improved throughput and

delay performance are achieved at the LLC layer using SLR at the RLC than using BLR.

The effect of slot-error rate P_s on the per-channel throughput and delay performance at the LLC is shown in Figs. 14 and 15, again for the same system parameter values in Fig. 12. Fig. 14 shows the throughput performance for $R_{\mathrm{llc}} \to \infty$ and for R_{rlc} taking values one and ∞ for both BLR and SLR. Note that $R_{\mathrm{rlc}} = 1$ and $R_{\mathrm{llc}} \to \infty$ corresponds to no error recovery at RLC at complete recovery at LLC. Also, $R_{\mathrm{rlc}} \to \infty$ corresponds to complete recovery at RLC (and no recovery at LLC). From Figs. 14 and 15, we can infer that SLR performs significantly better than BLR in terms of both throughput and delay.
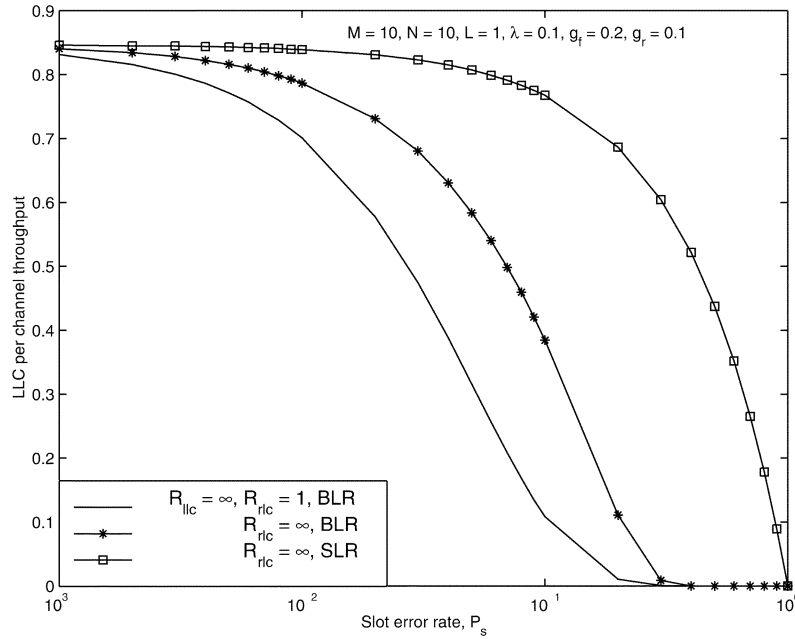
Fig. 14. LLC average per-channel throughput versus slot-error rate P_s. $N = 10$, $M = 10$, $L = 1$, $\lambda = 0.1$, $g_r = 0.1$, and $g_f = 0.2$.
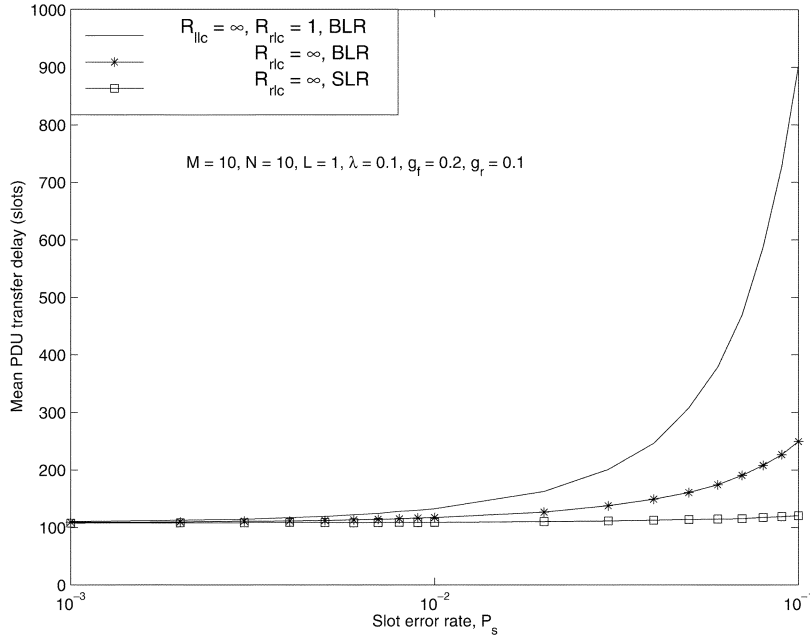

Fig. 15. Mean PDU transfer delay at LLC versus slot-error rate P_s. $N = 10$, $M = 10$, $L = 1$, $\lambda = 0.1$, $g_r = 0.1$, and $g_f = 0.2$.

From the average LLC throughput $\eta_{c,\text{llc}}$, shown in Figs. 12 and 14, we can compute the effective data rate at LLC in kilobits per second $\eta_{c,\text{llc}}^{(\text{kb/s})}$ using the relation

$$\eta_{c,\text{llc}}^{(\text{kb/s})} = \eta_{c,\text{llc}} \times \frac{n_{\text{inf}} \times \text{FL} - \text{OH}}{\text{FL} \times \text{BL} \times 4.615} \text{ kb/s} \quad (16)$$

where n_{inf} represents the number of information bits in each RLC block including the LLC header and checksum. FL represents the LLC frame length in terms of the number of RLC blocks, BL represents the RLC block length in terms of number of slots, and OH represents the number of overhead bits per LLC

frame. Note that the 4.615 in the denominator of the above expression accounts for the one slot duration, which is equivalent to one TDMA frame length of 4.615 ms. For the coding scheme CS-1, $n_{\text{inf}} = 181$ b for BLR and $n_{\text{inf}} = 40$ b for SLR. In our simulations, we set FL= 5 and BL= 4 for BLR and FL= 20 and BL= 1 for SLR. The overhead bits OH= 56 corresponds to LLC header and checksum bits. Note that $R_{\text{rlc}} \to \infty$ in Fig. 14 corresponds to full recovery in the RLC layer. This means that the per-channel throughput is the same at both RLC and LLC layers, but the effective data rates in kb/s are different. For SLR (see Fig. 14), the per-channel throughput at $P_s = 10\%$ is 0.77, which corresponds to an effective data rate of 6.19 kb/s at the
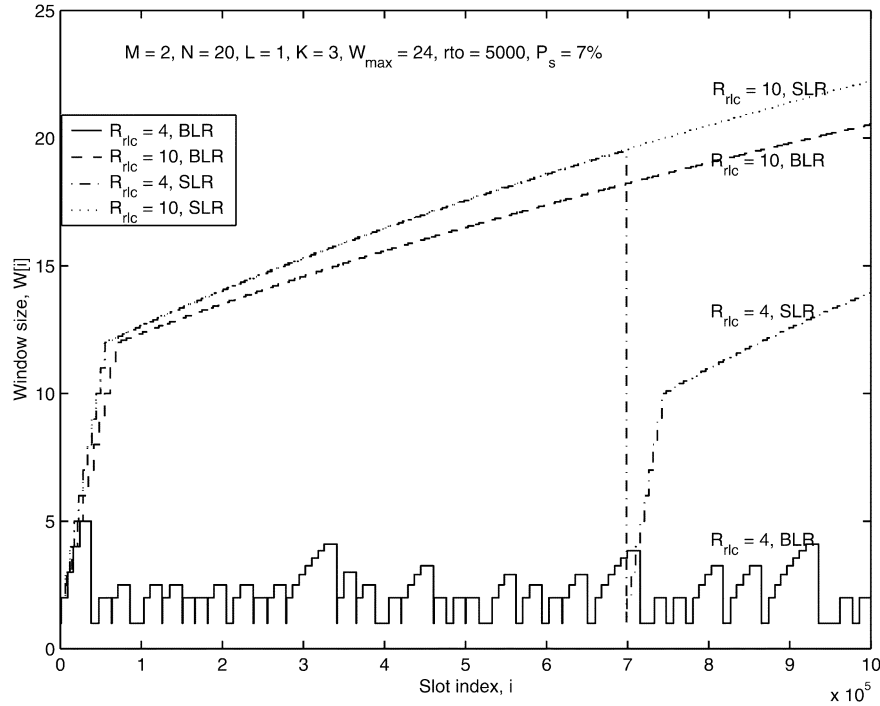
Fig. 16. Evolution of the TCP window size $W[i]$ versus the slot index i. $M = 2$, $N = 20$, $L = 1$, $R_{\mathrm{llc}} = 3$, $K_{\mathrm{TH}} = 3$, $P_s = 7\%$, and $W_{\max} = 24$ TCP packets. rto=5000 slots.

LLC layer. The corresponding effective data rate at the RLC layer is given by 6.68 kb/s.

B. TCP Performance Results

TCP is a well-known transport-layer protocol in the IP suite [4]. TCP is a reliable connection-oriented protocol that is widely used in popular applications such as http, ftp, telnet, etc. Several studies have analyzed the performance of TCP on wireless, but without considering ARQ in the link layer [5]–[7]. The performance of TCP with an RLP at the link layer in IS-95A code-division multiple-access (CDMA) system has been studied in [21]. In this section, we estimate the throughput performance of TCP on the GPRS uplink with the associated LLC/RLC/MAC layers. Being a connection-oriented protocol, TCP has call-setup, data-transfer, and call-clear phases. The bulk throughput performance of TCP is determined primarily by the data-transfer phase. In this study, we are interested in evaluating the TCP throughput in the data-transfer phase and follow the TCP data-transfer phase model described in [6].

We consider an ON–OFF traffic model, such as the web and e-mail traffic. Internet traffic typically exhibits a significant probability for long sessions and long interarrival times between sessions and packets. This leads to heavy-tailed (long-tailed) complementary cumulative distribution functions (ccdfs) for these typical measures [22]. Much work is done to find a suitable description of Internet traffic. A common approach is to approximate the empirical heavy-tailed ccdfs with *Pareto*, *Weibull*, *hyperexponential*, or *power law* distributions. Such a theoretical distribution allows the easy implementation of a generator that produces values of the described measure with a similar characteristic. In our simulations, to evaluate the performance of TCP over GPRS, we consider each TCP source to

generate a traffic pattern in which the interarrival time between packets (i.e., the OFF period) follows a Pareto distribution. The classical Pareto distribution with shape parameter β and location parameter a has the cumulative distribution function (cdf)

$$F(x) = P\{X \leq x\} = 1 - \left(\frac{a}{x}\right)^{\beta} . a; \quad \beta \geq 0; \quad x \geq a \tag{17}$$

with the corresponding probability density function (pdf) $f(x) = \beta a^{\beta} x^{-\beta-1}$. We model the ON period to consist of a geometrically distributed number of TCP packets. We assume that one TCP packet consists of five LLC frames and that each LLC frame contains 536 B. An LLC frame is segmented into 25 RLC blocks, each four slots in size. An erroneous RLC block can be retransmitted up to R_{rlc} times by the ARQ mechanism at the RLC layer. If the RLC layer cannot recover the erroneous blocks, they are passed on to the LLC layer. The ARQ mechanism at the LLC layer then tries to recover erroneous LLC frames by retransmission up to R_{llc} times. TCP takes care of the unrecovered erroneous LLC frames. We simulated 20 TCP sessions from mobile users (i.e., $N = 20$), each sending TCP packets to some hosts in an external packet-data network and evaluated the performance. The ON times of the sessions have a mean of five TCP packets. The Tahoe version of TCP with a fast retransmit threshold of $K_{\mathrm{TH}} = 3$ is used in the simulations [7].

Fig. 16 shows the evolution of TCP window size $W[t]$ as a function of time for a slot-error rate of $P_s = 7\%$, $N = 20$, $L = 1$, $K_{\mathrm{TH}} = 3$, $R_{\mathrm{llc}} = 3$, maximum window size of $W_{\max} = 24$ TCP packets, and a round trip timeout (rto) value of 5000

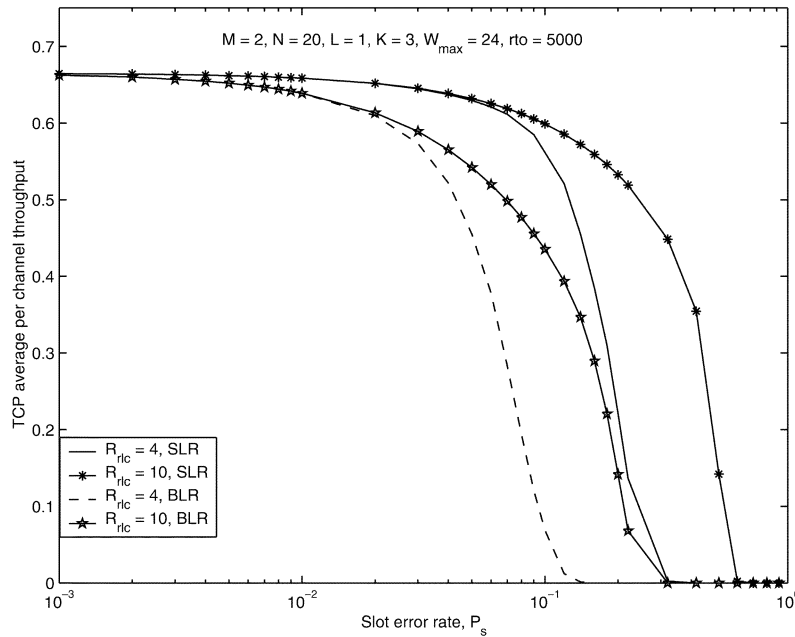Fig. 17. TCP throughput performance as a function of P_s and R_{rlc}. $M = 2$, $N = 20$, $L = 1$, $R_{\text{llc}} = 3$, $K_{\text{TH}} = 3$, and $W_{\max} = 24$ TCP packets. rto=5000 slots.

slots. The window evolution of four different cases are plotted as follows.

1) BLR with $R_{\text{rlc}} = 4$.
2) BLR with $R_{\text{rlc}} = 10$.
3) SLR with $R_{\text{rlc}} = 4$.
4) SLR with $R_{\text{rlc}} = 10$.

In Fig. 16, a comparison between $R_{\text{rlc}} = 4$ versus $R_{\text{rlc}} = 10$ for both BLR and SLR indicates that the TCP window size is more open for $R_{\text{rlc}} = 10$ than $R_{\text{rlc}} = 4$. This is because for $R_{\text{rlc}} = 4$ the recovery of erroneous blocks can be incomplete, which can result in more TCP timeouts and fast retransmits, which shrinks the TCP window size to 1. Since larger instantaneous window widths are good for achieving higher throughput, the choice of parameter value $R_{\text{rlc}} = 10$ is preferred over $R_{\text{rlc}} = 4$. Also, a comparison between the window evolutions for BLR and SLR with $R_{\text{rlc}} = 10$ reveals that the SLR results in larger instantaneous window widths as compared to BLR. This larger instantaneous window widths in SLR directly translate into higher TCP throughput compared to BLR, which is illustrated in Fig. 17. From Fig. 17, we observe that as R_{rlc} is increased, TCP throughput increases as expected. When R_{rlc} is large, more erroneous blocks are retransmitted and recovered in the RLC layer itself, rather than leaving them to LLC or TCP to recover by retransmitting the LLC frame or the entire TCP packet. From the TCP throughput values plotted in Fig. 17, $\eta_{c,\text{tcp}}$, we can compute the effective data rate in kb/s at the TCP layer $\eta_{c,\text{tcp}}^{(\text{kb/s})}$ using the relation

$$\eta_{c,\text{tcp}}^{(\text{kb/s})} = \eta_{c,\text{tcp}} \times \frac{n_{\text{inf}} \times \text{PL} \times \text{FL} - \text{OH}}{\text{PL} \times \text{FL} \times \text{BL} \times 4.615} \text{ kb/s} \quad (18)$$

where n_{inf} represents the number of information bits in each RLC block including the IP/LLC headers and checksum, PL represents the TCP packet length in terms of number of LLC frames, FL represents the LLC frame length in terms of the

number of RLC blocks, BL represents the RLC block length in terms of number of GSM slots, and OH represents the total number of overhead bits per TCP packet (IP header, LLC header/checksum). The 4.615 in the denominator of the above expression accounts for the one-slot duration that is equivalent to one TDMA frame length of 4.615 ms. For the coding scheme CS-1, $n_{\text{inf}} = 181$ for BLR and $n_{\text{inf}} = 40$ for SLR. In our simulations, we set PL=5, FL=25, and BL=4 for BLR and PL=5, FL=100, and BL=1 for SLR. The number of OH bits per TCP packet is 376 (320 IP header bits + 56 LLC overhead bits). In Fig. 17, for $P_s = 20\%$ the TCP throughput is 0.1021 for BLR, which corresponds to an effective date rate of 0.93 kb/s at the TCP layer. For the same system parameters, for SLR, the TCP throughput is 0.7087, which corresponds to an effective date rate of 6.02 kb/s at the TCP layer. Thus, there is a substantial improvement in the effective data rate at the TCP layer as we go from block level to SLR at the RLC, particularly for moderate-to-high channel-error rates.

V. CONCLUSION

We analyzed the performance of various layers (TCP/LLC/RLC/MAC) of the GPRS protocol stack. Using the theory of Markov chains, we derived the throughput and delay performance of the GPRS MAC protocol. We showed that the SLR scheme at the RLC layer performs better than the BLR defined in the standard. We investigated the interaction between the ARQs at the RLC and LLC layers; in particular, we studied the effect of the choice of the maximum number of retransmission attempts allowed at the RLC and LLC layers. Our performance results showed that it is more beneficial to do error recovery by allowing more retransmissions at the RLC layer than at the LLC layer. We also evaluated the performance of TCP with BLR and SLR at the RLC layer and showed that TCP performed better with SLR than compared to with

BLR. As further investigation, the performance analysis can be further extended for correlated channel errors, multislot operation in GPRS, and EGPRS. Performance analysis on the downlink is another topic for further investigation.

REFERENCES

[1] G. Brasche and B. Walke, "Concepts, services, and protocols of the new GSM phase 2+ general packet radio service," *IEEE Commun. Mag.*, vol. 35, pp. 94–104, Aug. 1997.

[2] C. Bettstetter, H. J. Vögel, and J. Eberspächer. (1999) GSM phase 2+ general packet radio service GPRS: Architecture, protocols, and air interface. *IEEE Commun. Surv.* [Online], pp. 2–14. Available: http://www.comsoc.org/livepubs/surveys/public/3q99issue/bettstetter.html

[3] A. Furuskar, S. Mazur, F. Muller, and H. Olofsson, "EDGE: Enhanced data rates for GSM and TDMA/136 evolution," *IEEE Pers. Commun. Mag.*, vol. 6, pp. 56–66, June 1999.

[4] W. R. Stevens, *TCP/IP Illustrated*. Reading, MA: Addison-Wesley, 1994, vol. 1.

[5] T. V. Lakshman and U. Madhow, "The performance of TCP/IP for networks with high bandwidth-delay products and random loss," *IEEE/ACM Trans. Networking*, vol. 5, pp. 336–350, June 1997.

[6] A. Kumar, "Comparative performance analysis of versions of TCP in a local network with a lossy link," *ACM/IEEE Trans. Networking*, vol. 6, pp. 485–498, Aug. 1998.

[7] M. Zorzi, A. Chockalingam, and R. R. Rao, "Throughput analysis of TCP on channels with memory," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1289–1300, July 2000.

[8] J. Cai and D. J. Goodman, "General packet radio service in GSM," *IEEE Commun. Mag.*, vol. 35, pp. 122–131, Oct. 1997.

[9] G. Brasche and B. Walke, "Analysis of multi-slot MAC protocols proposed for the GSM phase 2+ general packet radio service," in *Proc. IEEE Vehicular Technology Conf. (VTC'97)*, 1997, pp. 1295–1300.

[10] S. Hoff, M. Meyer, and A. Schieder, "A performance evaluation of internet access via the general packet radio service of GSM," in *Proc. IEEE Vehicular Technology Conf. (VTC'98)*, vol. 3, May 1998, pp. 1760–1764.

[11] Y. Argyropoulos, A. Bigloo, and J. van Peursem, "GPRS delay and capacity analysis for web browsing application," in *Proc. Int. Conf. Telecommunications*, vol. 2, 1998, pp. 32–36.

[12] C. Demeterscu, "LLC-MAC analysis of general packet radio service in GSM," *Bell Lab.s Tech. J.*, pp. 37–50, 1999.

[13] R. Kalden, I. Meirick, and M. Meyer, "Wireless internet access based on GPRS," *IEEE Pers. Commun. Mag.*, vol. 7, pp. 8–18, Apr. 2000.

[14] K. Premkumar and A. Chockalingam, "Performance analysis of RLC/MAC protocol in general packet radio service," in *Proc. National Conf. Communications (NCC'01)*, Kanpur, India, Jan. 2001, pp. 173–177.

[15] Digital cellular telecommunications system (phase 2+); General packet radio service (GPRS); mobile station-serving GPRS support node (MS-SGSN) logical link control (LLC) layer specification (GSM 04.64). European Telecommunications Standards Institute (ETSI) TC-SMG GPRS ad hoc, Sophia Antipolis, France. [Online]. Available: http://www.etsi.org/

[16] Digital cellular telecommunications system (phase 2+); General Packet radio service (GPRS); mobile station (MS)-BSS interface; radio link control/medium access control (RLC/MAC) protocol (GSM 04.60). European Telecommunications Standards Institute (ETSI) TC-SMG GPRS ad hoc, Sophia Antipolis, France. [Online]. Available: http://www.etsi.org/

[17] M. Zorzi and R. R. Rao, "Capture and retransmission control in mobile radio," *IEEE J. Select. Areas Commun.*, vol. 12, pp. 1289–1298, Oct. 1994.

[18] M. Zorzi, "Capture probabilities in random-access mobile communications in the presence of Rician fading," *IEEE Trans. Veh. Technol.*, vol. 46, pp. 96–101, Feb. 1997.

[19] Evaluation criteria for the GPRS radio channel Tdoc 56/95. European Telecommunications Standards Institute (ETSI) SMG2 GPRS ad hoc, Sophia Antipolis, France. [Online]. Available: http://www.etsi.org/

[20] R. W. Wolff, *Stochastic Modeling and the Theory of Queues*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[21] A. Chockalingam and G. Bao, "Performance of TCP/RLP protocol stack on correlated Rayleigh fading DS-CDMA links," *IEEE Trans. Veh. Technol.*, vol. 49, pp. 28–33, Jan. 2000.

[22] V. Paxson and S. Floyd, "Wide-area traffic: The failure of poisson modeling," *IEEE/ACM Trans. Networking*, pp. 226–244, June 1995.

K. Premkumar was born in Chennai, India, on December 27, 1975. He received the B.E. degree in electronics and communication engineering from the Alagappa Chettiar College of Engineering and Technology, Karaikkudi, India, in 1997 and the M.S. (research) degree in electrical communication engineering from the Indian Institute of Science, Bangalore, in 2002.

From August 2001 to September 2003, he was with Motorola India, Bangalore, where he was involved in the design and development of 3G systems and multiuser receivers. Since September 2003, he has been with the Indian Institute of Science, doing research in MIMO multiuser information theory. His research interests include the performance analysis of wireless communications and networks, power control, transmit diversity, and multiuser information theory.

A. Chockalingam (S'93–M'95–SM'98) received the B.E. (Hons.) degree in electronics and communication engineering from the P. S. G. College of Technology, Coimbatore, India, in 1984, the M.Tech. degree (with a specialization in satellite communications) from the Indian Institute of Technology, Kharagpur, in 1985, and the Ph.D. degree in electrical communication engineering (ECE) from the Indian Institute of Science (IISc), Bangalore, India, in 1993.

From 1986 to 1993, he was with the Transmission R & D Division, Indian Telephone Industries Ltd., Bangalore. From December 1993 to May 1996, he was a Postdoctoral Fellow and an Assistant Project Scientist in the Department of Electrical and Computer Engineering, University of California, San Diego (UCSD). From May 1996 to December 1998, he was in the Systems Engineering Group, Qualcomm Inc., San Diego, CA, as a Staff Engineer/Manager. In December 1998, he joined the Faculty in the Department of Electrical Communication Engineering, IISc, where he is an Associate Professor, working in the area of wireless communications and networks. He was a Visiting Faculty with UCSD during the summers of 1999–2002.

Dr. Chockalingam is a Recipient of the Swarnajayanti Fellowship from the Department of Science and Technology, New Delhi, India. He is an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.