Proceedings of the 2002 Winter Simulation Conference
E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, eds

SIMULATION ANALYSIS OF RLC TIMERS IN UMTS SYSTEMS

Xiao Xu
Yi-Chiun Chen
Hua Xu
Eren Gonen

Global Telecom Solutions Sector
Motorola, Inc.
1501 West Shure Drive
Arlington Heights, IL 60004, U.S.A.

Peijuan Liu

Department of Electrical & Computer Engineering
Northwestern University
2145 Sheridan Road
Evanston, IL 60208, U.S.A.

ABSTRACT

Radio Link Control (RLC) is the layer two protocol used in 3G UMTS cellular systems for flow control and error recovery. Due to the complexity of the protocol and the multitude of parameter configurations available, it is extremely difficult to model RLC analytically. Therefore we present a simulation model to study RLC performance in UMTS systems. We focus on the impacts of the poll prohibit timer and the poll timer on RLC throughput, goodput and delay. Our simulation results provide some insight into the optimization of these two timer values.

1 INTRODUCTION

Analog cellular systems, commonly referred to as the first generation systems, and digital cellular systems, the second generation (2G) systems, have been widely deployed and enabled wireless voice communications for decades. In recent years the demand for data services such as text messaging and wireless internet access has increased significantly. As a result, data communications become vital in the third generation (3G) systems to be deployed in the near future. UMTS (Universal Mobile Telecommunication System) is the evolved 3G system for GSM (Global System for Mobile Communications), which is currently the most widely deployed 2G cellular systems in the world. Compared to GSM, UMTS offers significant improvement in network capacity and data transmission rate and therefore facilitates a great variety of new packet data applications.

Since the air interface is the most critical resource in cellular systems, and RLC (Radio Link Control) is the major radio interface protocol of UMTS systems, we focus on RLC layer performance in our analysis. Packet data service protocols such as TCP, that are originally designed for faster and more reliable wire line communication links, are not well suited to handle the error-prone wireless air interface. RLC is designed to improve the performance of data traffic over the air interface so that higher layer protocols such as TCP can function effectively.

We devote special attention to two RLC timers, namely the poll prohibit timer and the poll timer. RLC implements multiple-reject ARQ (Automatic Repeat Request) for its flow control and error recovery through retransmissions. The poll prohibit timer and the poll timer play important roles in the poll triggering and status report transmission processes and have significant impacts on RLC performance in terms of throughput, goodput and delay. Therefore fine-tuning their settings is essential to UMTS network operators. In this paper we provide some insight into the optimization of these two timer values.

The paper is organized as follows: Section 2 provides an overview of the RLC protocol and its configurations. Section 3 describes the simulation model and parameter settings. Section 4 presents the simulation results and analysis. Final remarks and discussions of future work are addressed in Section 5.

2 RLC PROTOCOL OVERVIEW

In the layered architecture of UMTS radio interface protocols, RLC sits above the MAC (Medium Access Control) layer, which handles the scheduling of radio bearers with different QoS requirements, and below the RRC (Radio Resource Control) layer, which is responsible for setting up, modifying and releasing all the lower layer protocol entities. The specifications of the RLC protocol are in (3GPP TS 25.322 2001) and (Holma and Toskala 2000).

The functions of RLC cover both the control plane and the user plane of UMTS. Since our focus is on the user bearer traffic performance, the control plane functions, such as ciphering, protocol suspension, recovery, etc, are not included in the simulation model. There are three operational modes for RLC in UMTS, namely, transparent mode, unacknowledged mode, and acknowledged mode. The acknowledged mode is designed to provide reliable transport for packet data. It is the default mode for handling interactive and background applications, which are the applications of interest to us. Therefore we assume the acknowledged mode is the RLC operational mode throughout this paper and all the following discussions about RLC functions, configurations and parameters refer to the acknowledged mode of RLC.

As an overview, the basic RLC functions that are implemented in our simulation model are:

- **Segmentation and reassembly** – Higher layer data packets, or SDUs (Service Data Units) are segmented into RLC PDUs (Protocol Data Units). The RLC PDU size is set according to the lowest possible bit rate for the services using RLC. In case of small SDUs, several SDUs can be concatenated into one RLC PDU. At the receiving end, the PDUs that contain fragments of an SDU are reassembled and the SDU is delivered to the higher layer.
- **Transfer of User Data** – Transfer of data by RLC is controlled by QoS settings at the higher layers and scheduled at the MAC layer.
- **Error Correction** – Multiple-reject ARQ as described in (Chen et al. 2002) and (Yoon, Park and Min 2000) is used for retransmission in the event of erroneous or missing PDUs. It is a refined approach of the well-know selective-reject ARQ as described in (Stallings 1998) and (Fantacci 1996). The receiver sends a bitmap status report to the sender when it is polled. Inside the bitmap status report, the reception status (i.e., either ACK or NAK) within the receiving window and up to the last received PDU is listed. Several options of polling as well as status reports triggering are available and will be discussed in further details later.
- **In-sequence Delivery of SDUs** – This function preserves the order of SDUs, which are submitted for transfer by RLC from the sender to the receiver.
- **Duplicate Detection** – This function detects duplicate received RLC PDUs and ensures that the reassembled SDU is delivered only once to the higher layer.
- **Sliding Window Flow Control** – This function allows the receiver to control the rate at which the sender may send the PDUs. A receiving window size and a transmitting window size are negotiated between the receiver and sender during call setup. Unlike TCP's slow start with varying window size according to (Stevens 1994), the window size of RLC is fixed during the entire call unless a new window size is re-negotiated between the receiver and the sender. This is also part of the multiple-reject ARQ functions.

With multiple-reject ARQ, the RLC protocol provides a reliable service through retransmission to packet data applications over UMTS networks in the presence of high air interface bit error rates. In case of erroneous or lost PDUs retransmission is conducted by the sender upon reception of a status report from the receiver. A status report is sent either when the receiver is polled by the sender or self-triggered. For the sender, a polling request is made by marking the poll bit in the header of an outgoing RLC PDU. The possible triggers and inhibitors of polling are listed as follows:

- **Last PDU in Buffer** – The poll bit is set when the last PDU in the transmission buffer is sent.
- **Last PDU in Retransmission Buffer** – The poll bit is set when the last PDU in the retransmission buffer is sent.
- **Expiry of Poll Timer** – A poll timer is started when a PDU with the poll bit set is sent. If a status report is received before the timer expires the timer is cancelled. If the timer expires and no status reports have been received, a PDU with the poll bit set is sent.
- **Window Based Polling** – A status variable J, expressed in percentage terms, is maintained by RLC. It is calculated by dividing the number of PDUs sent within the current transmission window by the transmission window size. When J exceeds a certain percentage threshold, a PDU with the poll bit set is sent.
- **Periodic Polling** – A PDU with the poll bit set is sent periodically.
- **Poll Prohibit Timer** – Under the circumstances where several poll triggering options are present simultaneously in a system, a potential risk is that the network could be overwhelmed by excessive polling and status reports sent over the air interface. In WCDMA (Wideband CDMA), which is the air interface technology for UMTS, that translates to excessive power consumption and subsequently high level of interference to other users and reduction of overall system capacity. According to (3GPP TS 25.322 2001), one mechanism called the poll prohibit timer can be implemented to deal with this problem of excessive polling and status report transmission. At the transmitter, the poll prohibit timer is started once a PDU with the poll bit set is sent. No polling is allowed until this timer expires. If multiple polls were triggered during the period when this timer was in effect, only one poll is transmitted upon expiry of the timer.

As described above, RLC can be quite complicated with the large number of parameters that need fine-tuning. On the one hand, the UMTS operator has a great amount of flexibility with such a multitude of parameter settings to choose from; on the other hand, optimizing these parameters presents a significant challenge given the complex interactions among them.

The RLC parameter optimization is important, since inappropriate setting could deteriorate RLC performance or even cause deadlocks, as pointed out by (Chen et al. 2002). Unfortunately, there is very limited literature in this area besides (Chen et al. 2002), which provides guidance on choosing the optimal RLC window size and polling periods, and (Lefevre and Vivier 2001), which analyses the maximum number of retransmissions allowed (MaxDAT) and RLC buffer size and their interaction with TCP. In the subsequent 'Results and Discussion' section, we will present our findings about the poll prohibit timer and the poll timer.

3 SIMULATION MODEL

Figure 1 illustrates a UMTS network that is simulated in this study. The network nodes include SGSN (Serving GPRS Support Node), GGSN (Gateway GPRS Support Node), MSC (Mobile Switching Center), RNC (Radio Network Controller), Node B (Base Station), and UE (User Equipment). In addition, there are Iu-PS, Iu-CS, Iub, and Uu interfaces connecting those nodes in the UMTS network. The simulation modules have been developed as a framework to conduct analysis on all the major performance aspects at the TCP, RLC and MAC layers. But the scope of this paper is to cover only RLC performance issues related to two timers, i.e., the poll prohibit timer and the poll timer.

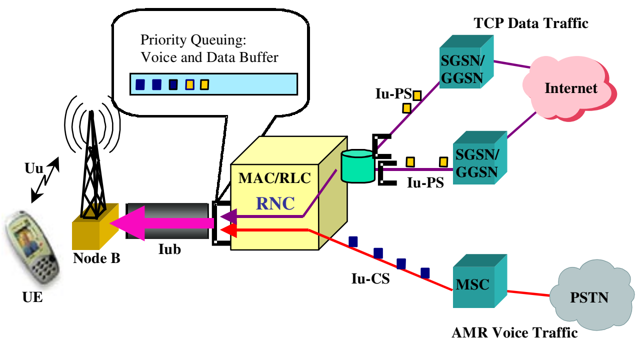

Figure 1: A Typical UMTS Network

As Figure 1 shows, voice traffic, which uses the standard AMR (Adaptive Multiple Rate) speech coding technique as specified by (3GPP TS 26.101 2001), comes from the PSTN (Public Switched Telephone Network). Data traffic, which uses TCP/IP, comes from the Internet. After core network switching nodes including GGSN, SGSN or MSC, data traffic and voice traffic are fed into the RNC, in which the RLC and MAC reside. Voice traffic is transparent to the RLC and has higher priority than data traffic. For data traffic, all the data flows are assumed to have the same priority. Each flow has a pair of transmission and retransmission buffers, of which retransmission has higher priority. The MAC scheduler schedules all the data flows based on the available resources, including air interface power and Iub link bandwidth. The scheduled data traffic then joins voice traffic for transmission over the Iub link. After reaching the Node B, both data and voice traffic will then be sent over the Uu air interface to the UE.

The simulation was developed with OpNet Modeler from OpNet Technologies, Inc., which is one of the most popular event-driven network simulators used by the telecommunications and networking industries for system development. The details of the major components in the simulation are described in the following.

- **TCP Data Traffic Source** – The TCP data traffic source is simulated as a Markov Modulated Poisson Process (MMPP) based on (Anagnostou, Sanchez-P and Venieries 1996). The active periods and idle periods are exponentially distributed. No packets are generated during the idle periods. During the active periods, packet inter-arrival times and packet sizes follow exponential distributions.
- **AMR Voice Traffic Source** – The AMR voice traffic is simulated as a two-state ON-OFF Markov process. In the ON and OFF states, the voice frame sizes and their patterns are generated according to (3GPP TS 26.101 2001).
- **MAC/RLC** – The MAC and RLC functions are modeled in a single simulation module to facilitate better interaction between these two layers. The RLC functions implemented in the simulation are detailed in Section 2 above. The function of MAC is to schedule among voice users and data users based on the air interface power and Iub link bandwidth constraints.
- **Iub Link** – In practice, ATM protocol is used for the Iub link in UMTS. In the simulation, however, the Iub is modeled as a direct link with priority queuing between voice and data packets.
- **Uu Air Interface** – The air interface power consumption for voice and data packets using different WCDMA spreading factors is modeled as a Gaussian distribution. The mean and variance of this Gaussian distribution are extracted from statistical results of a Motorola proprietary simulator. We considered air interface error rates of 0%, 10% and 20%.
- **UE** – The UE acts as the peer entity to the RLC in RNC. For downlink, it terminates the voice and data packets from the RNC and Node B. For uplink, it also generates voice and data traffics to the Node B and RNC according to the TCP data traffic source and the AMR voice traffic source described above.

4 RESULTS AND DISCUSSION

4.1 Simulation Configuration and Metric Definition

We simulated 60 AMR voice users at 12.2 kbps and 16 data users at 100 kbps. At the Iub link, we provided an E1 link with 2 Mbps. The Iub link buffers were adjusted to sufficient capacities in the simulation such that packet loss at the Iub link was minimal.

Compared with an analytical approach as in (Yoon, Park and Min 2000), the simulation is capable of exploring the system behavior in the presence of very complex parameter settings. In our case, we implemented all of the RLC polling mechanisms specified by (3GPP TS 25.322 2001). Table 1 shows the specific RLC polling settings in the simulation. We disabled Periodic Polling because in the presence of the poll prohibit timer, the periodic timer is no longer able to trigger polling periodically. (Please note that the values of the poll timer and the poll prohibit timer will not be fixed at the same time. When we study the effects of the poll timer, the poll prohibit timer is set at 0.1 second; when we study the effects of the poll prohibit timer, the poll timer is set at 0.5 second.)

Table 1: RLC Parameter Settings

Last PDU in Buffer	Enabled
Last PDU in Retransmission Buffer	Enabled
Periodic Polling	Disabled
Window Based Polling Threshold Value	90%
Poll Timer Value	0.5 sec
Poll Prohibit Timer Value	0.1 sec

With several poll triggering mechanisms interacting with each other, it is almost impossible to derive RLC behavior analytically, but with our simulation we were able to present some noteworthy trends for this setting.

In our results, we focus on three RLC performance metrics: data throughput, data goodput and RLC SDU delay. Throughput (kbps) is defined as the total bit rate measured at the Iub that is used for transmitting data traffic. This includes both transmission of data packets for the first time and retransmission. Goodput (kbps) is measured at the UE for the received PDUs, which carry "good" information for reassembly of an SDU. Therefore, neither erroneous PDUs nor duplicate PDUs are counted into the goodput. In our simulation, since the amount of data generated at the TCP data traffic source (i.e., the offered load) is fluctuating from scenario to scenario, it is difficult and meaningless to compare absolute values of throughputs and goodputs for different scenarios. To solve this problem, we normalized throughputs and goodputs with respect to their offered loads in each scenario. We express normalized throughputs and goodputs in percentage terms of their offered loads in our analysis. Since throughput also counts in retransmissions, its value may exceed 100% of the offered load.

The RLC SDU delay is from the time at which the higher layer delivers the packets to RLC at the transmitter till the time at which RLC relays the correctly reassembled packets to the higher layer at the receiver. The RLC SDU delay consists of: queueing delay and MAC scheduling delay at RNC, queueing delay and transmission delay at the Iub link, processing delay at Node B, transmission delay over the Uu air interface, and reassembly delay at UE. In the simulation, the Node B processing delay and the Uu air interface transmission delay together are set to a fixed 60ms for all the PDUs according to (3GPP TS 25.853 2001).

In (Chen et al. 2002), the impact of RLC window size was studied in detail. The major findings are: a large RLC window size will generally increase throughput and goodput and reduce RLC SDU delay. When window size is small, RLC is not able to utilize the bandwidth resources available to it and SDU delay also suffers from this low utilization. After window size surpasses a threshold, this restriction posed by window size no longer exists. Since we are using very similar configurations for our model, we are able to use the window size threshold values indicated by (Chen et al. 2002) for the benefit of our analysis. For our simulations, we set a sufficiently large window size of 128 PDUs.

4.2 Poll Prohibit Timer

Figure 2 and Figure 3 present the effects of poll prohibit timer on throughput and goodput, respectively. The two figures demonstrate that if the value of the poll prohibit timer is very large, i.e., polling is prohibited for a long time, throughput and goodput will be low. In multiple-reject ARQ, status reports are essential for advancing the transmission window. A large poll prohibit timer value implies infrequent polling and therefore infrequent status reports. The RLC transmission window may stall frequently in this case. Furthermore, the throughput and goodput degradation is more severe at higher error rates because polling requests and status reports are more prone to loss over the air, thereby making the effect of a large poll prohibit timer more pronounced.

When poll prohibit timer value is sufficiently small, RLC is able to accommodate all the offered load since the goodput values at the left part of Figure 3 are 100%. But when poll prohibit timer value is extremely small, there is an increase in throughput as shown in Figure 2. At such small poll prohibit timer values, RLC could experience excessive polling and unnecessary retransmissions before the arrival of the requested status reports. These retransmissions account for the up-tick of throughputs in the extreme left part of Figure 2.

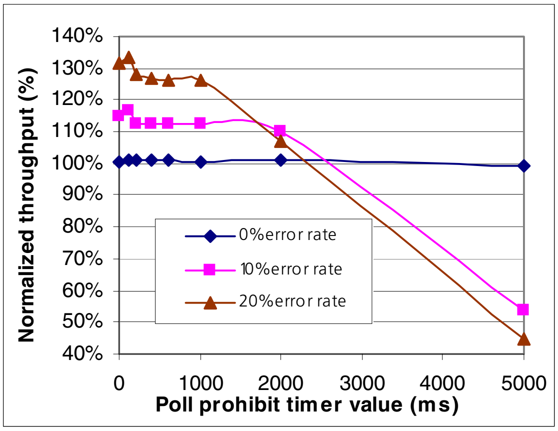

Figure 2: Throughput vs. Poll Prohibit Timer Value

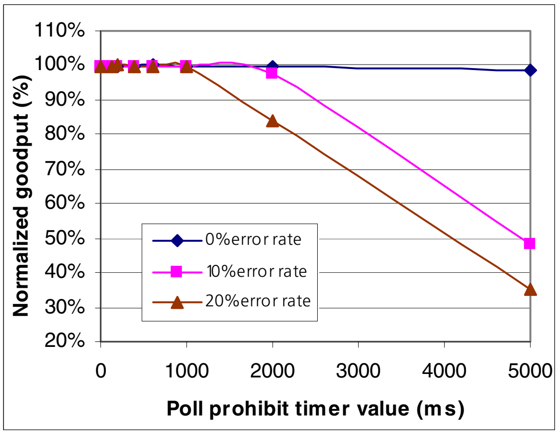

Figure 3: Goodput vs. Poll Prohibit Timer Value

Figure 4 illustrates the effects of poll prohibit timer on RLC SDU delay. When the goodputs in Figure 3 are well below 100%, the RLC SDU delays approach infinity as some RLC PDUs never get transmitted. With adequate polling frequencies (i.e., small poll prohibit timer values), RLC SDU delays are finite and small. Also with the same poll prohibit timer value, the higher the air interface error rate, the longer the RLC SDU delay. This is because at higher error rates, more retransmissions are needed before the successful reassembly of an RLC SDU.

In (Chen et al. 2002), the effects of periodic timer on throughput, goodput and RLC SDU delay are studied in a model where the periodic timer is the only poll trigger and the poll prohibit timer is disabled. Comparing the throughput, goodput and RLC SDU delay figures in (Chen et al. 2002) with Figures 2, 3 and 4, we can see that the shape and trend of the curves are identical. We believe the underlying rationale is that given a sufficiently large window size, and in the presence of multiple poll triggering mechanisms, whenever the poll prohibit timer expires, a poll will be sent. Therefore the poll prohibit timer is essentially acting like a periodic poll timer, and the length of the polling period is exactly the poll prohibit timer value.

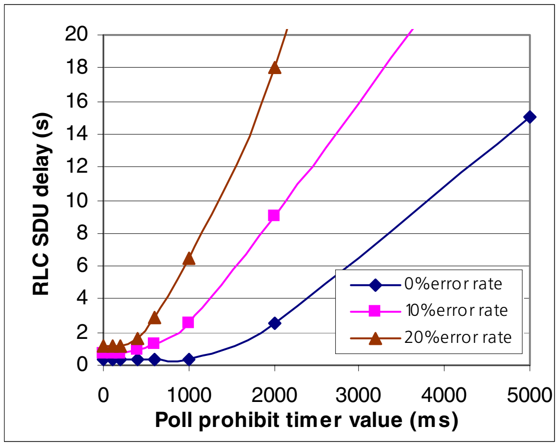

Figure 4: RLC SDU Delay vs. Poll Prohibit Timer Value

Based on our observations of Figures 2, 3 and 4, we find that setting the poll prohibit timer value to be 100ms is optimal in the sense that SDU delay is small while throughput is not excessively high. Please note that due to the way throughput is defined in this paper, given that the goodput is 100% (all the offered load is accommodated), the lower the throughput, the smaller the waste of bandwidth and the better the RLC performance. Actually, any normalized throughput higher than 100% means there are errors and retransmissions.

More generally, we define the RLC round trip time (RTT) as the time it takes a PDU to arrive at the receiver plus the time it takes a status report to arrive back at the sender, in case no error or retransmission occurs. Further simulation results points out that the ratio of poll prohibit timer value to RLC RTT has an optimal performance range of between 0.8 and 1.2.

4.3 Poll Timer

For our analysis of the poll timer, we fixed the poll prohibit timer at 100ms based on the above results of the poll prohibit timer. Figure 5 illustrates the effects of the poll timer on throughput. At small poll timer values, throughputs are high. As we increase the poll timer value, throughput decreases and it levels off after a certain poll timer threshold. When poll timer values are small (especially smaller than the RLC RTT), the poll timer expires soon after a PDU with the poll bit set is sent. No associated status reports would arrive within this period of time since it is even shorter than the RLC RTT. In this scenario premature re-polling PDUs are sent, and a significant portion of those re-polling PDUs are unnecessary retransmission PDUs which increase the throughput with no benefit to goodput or SDU delays.

As the poll timer value surpasses the threshold, further increasing it no longer has any influence on the throughput.

That is because at large poll timer values, the significance of the poll timer is reduced. Before the poll timer expires, it is highly likely that the associated status report has already been received or that other poll triggering mechanisms such as Last PDU in Buffer, Last PDU in Retransmission Buffer or Window Based Polling will take effect. Therefore RLC is less likely to rely on the poll timer to trigger polling.

A second trend demonstrated by Figure 5 is that for poll timer values of less than 100ms, throughput values are the same. Since the poll timer, as well as other poll triggering mechanisms, is subject to the constraint of the poll prohibit timer. As noted before, the poll timer and the poll prohibit timer are always started simultaneously when a PDU with the poll bit set is sent. In our simulation, the poll prohibit timer value is 100ms. Setting the poll timer value to be less than that is equivalent to setting it at 100ms because even if it expires before 100ms, it has to wait for the poll prohibit timer to expire before polling can be triggered.

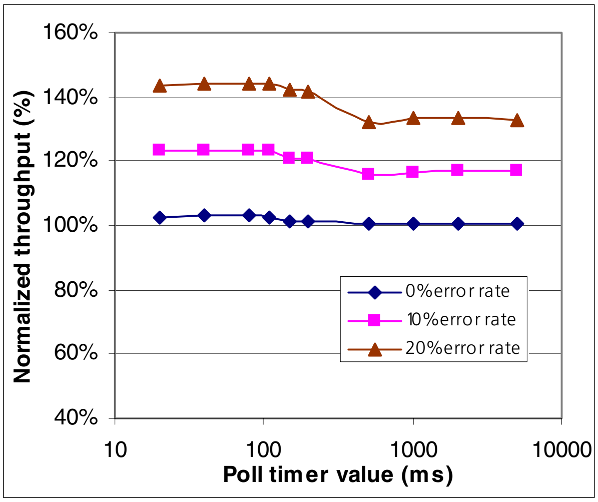

Figure 5: Normalized Throughput vs. Poll Timer Value

Figure 6 shows the effects of the poll timer on goodput. The three curves overlay at 100%. As we have provided a sufficiently large RLC window size and multiple poll triggering mechanisms, the RLC is able to accommodate all the offered load. Therefore goodput values are not sensitive to the poll timer.

The relationship between RLC SDU delays and poll timer values is displayed in Figure 7. At small poll timer values, the RLC SDU delays are small. As poll timer values increase, RLC SDU delays also increase. A small poll timer value helps the RLC to recover quickly from a lost or erroneous PDU by way of timely retransmissions. We notice that as we increase the value of the poll timer, the rate of increase for RLC SDU delays slows down. This is consistent with our earlier findings that the significance of the poll timer reduces as its value increases. As an extreme example, assume the poll timer value approaches infinity, i.e., there is no poll timer. Due to the existence of other poll triggering mechanisms, we still find the RLC SDU delay to be finite. (Simulation results are not shown.) Therefore the RLC SDU delays cannot keep increasing fast with increasing poll timer values.

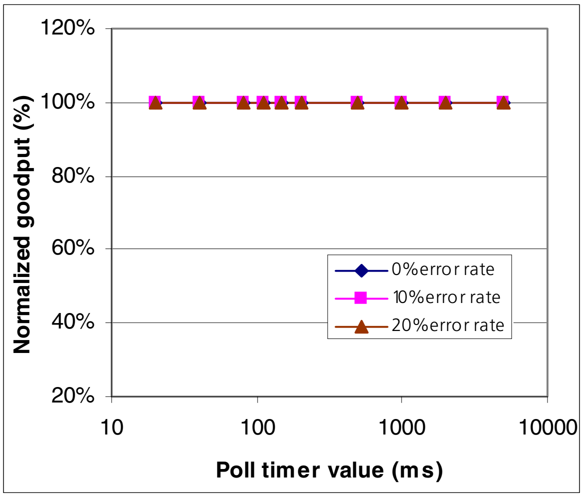

Figure 6: Normalized Goodput vs. Poll Timer Value

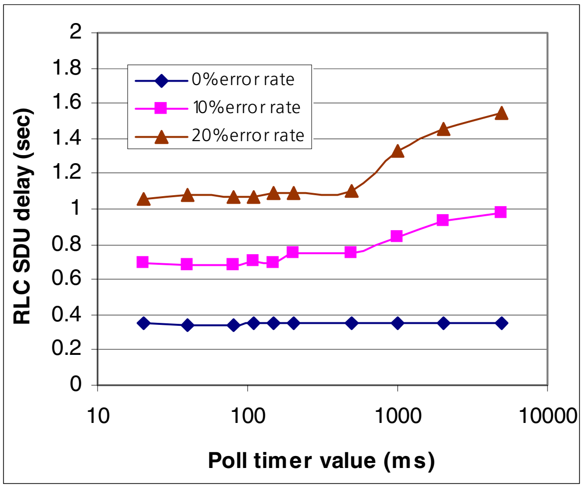

Figure 7: RLC SDU Delay vs. Poll Timer Value

Again, we notice the trend that for poll timer values smaller than 100ms, the RLC SDU delay curve is flat. This is consistent with our observations about the throughput curves. Therefore the poll timer value in general should not be set to be smaller than the poll prohibit timer value.

5 CONCLUSIONS AND FUTURE WORK

We have studied the RLC poll prohibit timer and poll timer using our simulation model. Our results suggest that there is a tradeoff between throughput and delay. Generally, small values for these two timers produce low delays but high throughputs (which could overload the WCDMA air interface), while large values for these two timers produce moderate throughputs but long delays. Unusually large values for the two timers should be avoided as they deteriorate RLC performance. We are able to identify in our simulation scenarios an optimal combination (0.1 second for the poll prohibit timer and 0.5 second for the poll timer), which results in a moderate throughput and a low delay.

In practice, a UMTS operator may not be able to identify such an optimal parameter combination. Under that circumstance, the traffic load sensitivity of the WCDMA air interface must be weighed against the delay requirements of data applications before a choice is made about these two parameters.

Further research will be conducted to study the RLC behavior under different higher layer traffic models e.g., WAP (Wireless Application Protocol), and the interaction between RLC and higher layer protocols including TCP and WAP.

REFERENCES

3GPP TS 25.322. 2001. 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; RLC protocol specification, version 4.3.0. Available online via `<http://www.3gpp.org> [accessed November 12, 2001]`.

3GPP TS 25.853. 2001. 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Delay budget within the access stratum, version 3.1.0. Available online via `<http://www.3gpp.org> [accessed November 12, 2001]`.

3GPP TS 26.101. 2001. 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; AMR speech codec frame structure, version 3.2.0. Available online via `http://www.3gpp.org [accessed November 12, 2001]`.

Anagnostou, M. E., Sanchez-P, J.-A., and Venieries, I. S. 1996. A multiservice user descriptive traffic source model. *IEEE Trans. On Communications* 44: 1243-1246.

Chen, Y., Xu, X., Xu, H., Liu, P., and Gonen, E. 2002. Simulation analysis of RLC for packet data services in UMTS systems. Submitted *to IEEE Wireless Communications and Networking Conference 2003*.

Fantacci, R. 1996, Queuing analysis of the selective reject automatic repeat protocol wireless packet networks. *IEEE Trans. on Vehicular Technology* 45: 258-264.

Holma H. and Toskala A. 2000. *WCDMA for UMTS,* New York: John Wiley & Sons.

Lefevre, F. and Vivier, G. 2001. Optimizing UMTS link layer parameters for a TCP connection. In *Proceedings of IEEE Vehicular Technology Conference,* 2318-2322.

Stallings, W. 1998. *High-speed Networks: TCP/IP and ATM Design Principles,* Upper Saddle River, NJ: Pretice Hall.

Stevens, W. R. 1994. *TCP/IP Illustrated, Volume I – The Protocols,* Reading, MA: Addison Wesley.

Yoon, U., Park, S. and Min, P. 2000. Performance analysis of multiple rejects ARQ at RLC (Radio Link Control) for packet data service in W-CDMA system. In *Proceedings of IEEE Global Communications Conference 2000,* 48-52.

AUTHOR BIOGRAPHIES

XIAO XU received his M.S. degree in Industrial Engineering and Management Sciences from Northwestern University, Evanston, IL, in 2000. Since 2001 he has been working at Motorola on wireless networks performance modeling and analysis. His main focus of work has been stochastic modeling , simulation analysis and network capacity planning of 3G UMTS cellular systems. His email address is `<xiao.xu@motorola.com>`.

YI-CHIUN CHEN received his M.S. and Ph.D. degrees in Electrical and Computer Engineering from Northwestern University, Evanston, IL, in 1994 and 2000 respectively. Since 1995, he has been with Motorola working on the performance analysis and system design of wireless communication systems. His research experience and interests include performance evaluation, traffic engineering, simulation modeling, capacity planning, system architecture, and algorithm design. His email address is `<y.chen@motorola.com>`.

HUA XU received her Ph.D. degree in Industrial Engineering from Georgia Institute of Technology, Atlanta, GA, in 1992. She has been working at Motorola on the performance analysis of digital cellular systems since. Her research interests include probability and stochastic modeling and simulation of communication system/network in wireless systems, and real-time traffic management and access control techniques. Her email address is `<hua.xu@motorola.com>`.

PEIJUAN LIU is a Ph.D. candidate in the Department of Electrical and Computer Engineering at Northwestern University, Evanston, IL. She was a Motorola Engineering Intern during the summers of 2000 and 2001. Her research interests include wireless resource allocation, stochastic modeling and its applications in communication networks. Her email address is `<peijuan@ece.nwu.edu>`.

EREN K. GONEN received his M.S. degree in Electrical and Computer Engineering from Georgia Institute of Technology, Atlanta, GA, in 2000. Since 2001, he has been with Motorola working on performance analysis of communication systems. His main focus of work has been stochastic modeling, simulation and performance analysis of 3G cellular networks based on UMTS. His email address is `<eren.gonen@motorola.com>`.