A New ARQ Protocol for Wireless ATM Networks

Ian F. Akyildiz and Inwhee Joe

Broadband and Wireless Networking Laboratory
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA
ian@ece.gatech.edu; inwhee@ece.gatech.edu

Abstract:
This paper describes the design and performance of a new ARQ protocol for wireless ATM networks. The wireless channel is characterized by a higher and variable error rate in comparison with fiber-based networks for which ATM was designed. The purpose of the protocol is to provide a capability to dynamically support ATM-based communications in a fluctuating transmission environment by using selective retransmission. The key ideas in the protocol design consist of variable packet size and periodic status message. The packet size is changed adaptively with the optimal size according to the time-varying conditions of the wireless channel, as a result maximizing the throughput efficiency. The proposed protocol has been validated using a software emulator which incorporates a wireless channel model. Experimental performance results based on the implementation are presented.

Key Words: Wireless ATM, Data Link, Survivability, Retransmission, Variable Packet Size, Periodic Status Message.

1 Introduction

The use of ATM over wireless links immediately brings up a fundamental issue in the way that ATM will be used. Because of the fading effects and interference, the wireless link is characterized by a higher and variable error rate when compared with fiber-based links for which ATM was designed. Such difference in error characteristics leads to a data link protocol using ARQ (Automatic Repeat Request) for wireless ATM networks, in order to insulate the ATM network layer from wireless channel impairments.

Recently, a data link protocol based on the ARQ scheme has been proposed for the radio channel [2]. However, it assumes circuit-mode data and furthermore it is quite complex to maintain various packet pointers. In addition, an asymmetric link protocol has been presented in [1] with low throughput. Here we propose an efficient data link protocol which assumes packet-mode data, attempting to provide ATM-based transmission in response to fluctuating lossy links by using selective retransmission.

In this paper, we focus on design and performance evaluation of the data link protocol. In the next section we discuss a detailed description of the protocol. In Section 3 we explain how to adapt the packet size to the channel conditions. In Section 4 we present performance evaluation results from a software emulation of the protocol. Finally we conclude the paper by highlighting our contribution and future research directions.

2 Protocol Description

The "variable packet size" and "periodic status message" are the key ideas in our new protocol, which minimizes the processing overhead and maximizes the throughput efficiency

2 (bits)	8	4	1		16	1
TI	seq	SC	HO	Payload (variable size)	CRC	EOT

(a) Data Packet Format

2 (bits)	8	4		16
TI	MSN	m	BMAP	CRC

(b) Control Packet Containing Receiver's State

Figure 1: Packet Formats

over wireless links with a high and variable error rate. The throughput efficiency is very sensitive to the packet size, i.e., when a packet size is too long, there is an increased need for retransmissions, while a small packet is inefficient because of the large overhead rather than the actual data. Therefore, the packet size should be chosen adaptively based on the time-varying conditions of the wireless channel, which will be discussed in the next section.

The idea of "periodic status message", where the receiver sends its status to the transmitter on a periodic basis as in the SNR protocol [3], avoids redundant retransmissions and simplifies the protocol by eliminating the timeout mechanism. Moreover, it removes the dependence on the error prone medium. If the receiver does not receive packets due to the channel errors, it can still send periodic status messages, because they are sent periodically by a local timer and not by the event of receiving a packet. Likewise, even if a status message gets lost, a subsequent message will always follow.

a) Data Packet Format

The format of the data packet is shown in Figure 1(a). *TI* is the traffic type indicator. Since the protocol supports a variety of multimedia traffic with different requirements, the *TI* field (e.g., 2 bits) is used to indicate whether a packet is data or control, and its traffic type such as quality critical (e.g., data and still images) or time critical traffic (e.g., voice and video). The data packets have a sequence number *seq* (e.g., 8 bits) in order to identify out-of-sequence packets at the receiver. *SC* is the segment counter (e.g., 4 bits) used for segmentation and reassembly of the ATM cells. The packing scheme for ATM cells is explained in Section 3.2 in detail. *HO* (e.g., 1 bit) is used to indicate the packets being sent during handoff. The *payload* field contains information bits with the variable size. The 2-byte *CRC* field provides error detection for the data packets. When there is no more data

0-7803-4788-9/98/$10.00 © 1998 IEEE.

to transmit, the end of text *(EOT)* bit [1] is set in the last packet.

b) Control Packet Format

Status messages are sent from the receiver to the transmitter through control packets on a periodic basis using a local timer. Especially when a packet with the EOT bit set is received, the receiver sends its status immediately without waiting until the next period to send status messages. This allows for a fast response when small amounts of data are being sent. The period of sending status messages should be chosen so as not to waste much bandwidth for control packets while at the same time offsetting the effect of a noisy channel.

As shown in Figure 1(b), the control packet has the following fields: TI is the traffic type indicator, MSN is the maximum sequence number of the packet below which every packet has been received correctly at the receiver, m denotes the current frequency of control packets, $BMAP$ is the bit map indicating packets outstanding between MSN and $MSN+WS-1$ where WS is the window size, and CRC is used for error detection.

c) Control Packet Transmissions

Let T_i be the time interval in which the control packets will be sent from the receiver.

$$T_i = max\{\frac{RTD(L_i)}{m_i}, \delta\}, \qquad i = 0, 1, 2, \ldots \quad (1)$$

where RTD is an estimate of the round trip delay depending on the variable packet length L_i, m_i is a parameter used to adjust the time interval of control packet transmission according to the channel condition, and δ is the average packet inter-arrival time on the wireless channel.

The initial value m_0 is determined by considering a tradeoff between bandwidth and response time at the worst case of the channel BER, thereby setting the initial conditions of m_i and channel BER. In order to adjust the value T_i to the varying channel condition, m_i will be recalculated periodically based on the following two factors: channel activity and channel BER. The calculation period is chosen to be RTD, because m_i is the value defined per RTD and hence can be reinitialized every RTD unit of time. For channel activity, one bit is used to indicate whether the channel is active or not by counting the number of packets received during the last RTD interval. If there are no packets arrived, the channel activity bit is set to 0. The ultimate goal of the T_i adjustment is to minimize the bandwidth to be wasted by control packets when the channel is idle or when the channel BER becomes better. In particular, when the channel condition improves, the frequency of control packets can be reduced by acknowledging as many as packets as possible at a time. In order to increase the control packet interval T_i when the channel is idle or the channel condition gets better, m_i is used to slow down the frequency of control packets, i.e.,

$$T_{i+1} = max\{\frac{RTD(L_{i+1})}{m_{i+1}}, \delta\} \quad (2)$$

where $m_{i+1} = m_i/2$. Since m_i represents how many control packets to send in a round trip delay, the minimum value of m_i will be one so that the transmitter can have at least one acknowledgment within a window interval for advancing the window.

On the other hand, activity on the channel or the degraded channel condition makes the control packet interval T_i return to the original state by restoring the initial value m_0. For time critical traffic, T_i can also be reduced to obtain a better response time. In that case, we should consider a tradeoff between the frequency of status messages and the expensive bandwidth.

For flow control, the sliding window mechanism is used in our protocol. The window size WS is measured in packets.

$$WS = max(\frac{RTD(L_i)}{T_{tx}(L_i)}) \quad (3)$$

Even if the round trip delay RTD and the transmission time T_{tx} depend on the packet length L_i, the window size will be fixed based on the worst case calculation, in order to make the implementation simpler. The buffer size should be larger than the window size to accommodate newly arriving data packets during multiple retransmissions at the transmitter and to take care of out-of-sequence packets at the receiver. Usually, the buffer size is a multiple of the window size. Once the window size is obtained from Eq. (3), the buffer size and the $BMAP$ size can be determined accordingly.

A state table for all outstanding packets is maintained at the transmitter, and updated as new control packets arrive from the receiver. The entries in the state table consist of *State[seq,count]*, where *seq* is the sequence number of an outstanding packet, and *count* is the retransmission count. When a packet is transmitted, the initial value of retransmission *count* for that packet is set to

$$count = \lceil m_i \rceil \quad (4)$$

where m_i comes from the latest received control packet. Every time a control packet is received, the transmit buffer is compared with the received $BMAP$ field. Packets which have been acknowledged are removed from the state table. At the same time, the retransmission *count* of all other packets in the state table is decreased by 1, because it is expressed in unit of the control time interval. A packet is retransmitted only if the retransmission *count* goes to zero. Since retransmission has priority over transmission of a new packet, the retransmission packet is sent immediately with the retransmission *count* reset to the initial value.

3 Packet Length Adaptability

In this section, we explain how to adapt the packet size to the channel conditions to maximize the throughput efficiency.

3.1 Packet Length Adaptability

There exists an optimal packet size in the sense of maximizing the throughput efficiency of the protocol. The optimal choice is found to depend on the error characteristics of the channel and on the number of overhead bits used for control. Since our protocol retransmits only packets in error based on the SR-ARQ scheme, the optimal information payload size is given by [6],

$$l_{opt} = \frac{-h\ln(1-P_b) - \sqrt{-4h\ln(1-P_b) + h^2\ln(1-P_b)^2}}{2\ln(1-P_b)} \quad (5)$$

where P_b is the channel BER and h is the number of overhead bits per packet. For an AWGN (Additive White Gaussian Noise) Rayleigh fading channel with binary DPSK (Differential Phase Shift Keying) modulation, the probability of bit error is given by [5],

$$P_b = \frac{1}{2(1+\gamma_b)} \tag{6}$$

where γ_b is the average SNR (Signal-to-Noise Ratio) per bit. For some channels the BER varies with time. This is particularly the case for wireless channels where signal fading and interference are unpredictable and time varying. As a result, the packet size should be variable according to the fluctuating channel conditions. Since the overhead size h is fixed, the optimal packet size is chosen based on the channel condition at a given time, maximizing the throughput efficiency of our protocol (SDLP).

For our SDLP protocol, the *throughput efficiency* η can be derived as,

$$\eta = (1-P_b)^{l+h} \cdot (\frac{l}{l+h}) \tag{7}$$

where P_b is the channel BER, l is the number of payload bits, and h is the number of overhead bits. As shown in Eq. (7), the throughput efficiency consists of two terms: the first term represents the throughput of the protocol, while the second term represents the transmission efficiency. In order to evaluate the throughput efficiency of our protocol for the variable packet size, now consider the *relative throughput efficiency* η_R:

$$\eta_R = \frac{\eta_l}{\eta_{opt}} \tag{8}$$

where η_l is the throughput efficiency for a fixed packet size l irrespective of channel error rates, and η_{opt} is the throughput efficiency for the variable packet size using optimal values with the channel error rates from Eq. (5). This expression points out how efficient the proposed protocol with the variable packet size is, as compared to the classical SR-ARQ protocol with a fixed packet size.

Given the 32-bit overhead as used in the data packet in Section 2.1, the optimal payload size l_{opt} from Eq. (5) is shown for different error rates in Figure 2(a) and for different SNR values in Figure 2(c), respectively. The throughput efficiency for the optimal packet size, η_{opt}, is also shown in Figures 2(b) and 2(d), which is the case of our SDLP protocol. For example, the optimal size is about 170 bits and its corresponding throughput efficiency is about 0.69 at a BER of 10^{-3} (SNR = 27 dB). As shown in Figure 3, choosing the optimal packet size at a given channel BER always gives the best throughput efficiency, while a much larger or smaller packet size would drop the throughput efficiency drastically. That is, the relative throughput efficiency is about 0.23 at the same BER of 10^{-3} above when the payload size is much larger (e.g., 1773 bits) than the optimal size. Similarly, when the payload size is chosen to be a much smaller value (e.g., 8 bits) than the optimal size based on the worst case BER such as 10^{-1}, the relative throughput efficiency is also as low as 0.28. In summary, since the wireless channel is time-varying, the packet size should vary adaptively to the optimal packet size with the channel conditions, so as to obtain the best throughput efficiency.

3.2 Packing Scheme for ATM Cells

The transmission unit at the link layer is called a "segment" and determined by considering the throughput efficiency of the protocol. As shown in Figure 2(b), the throughput efficiency decays rapidly in the range between the BERs 10^{-3} and 10^{-2}. In other words, the throughput efficiency at the BER 10^{-3} is 0.69, while at the BER 10^{-2} it is as low

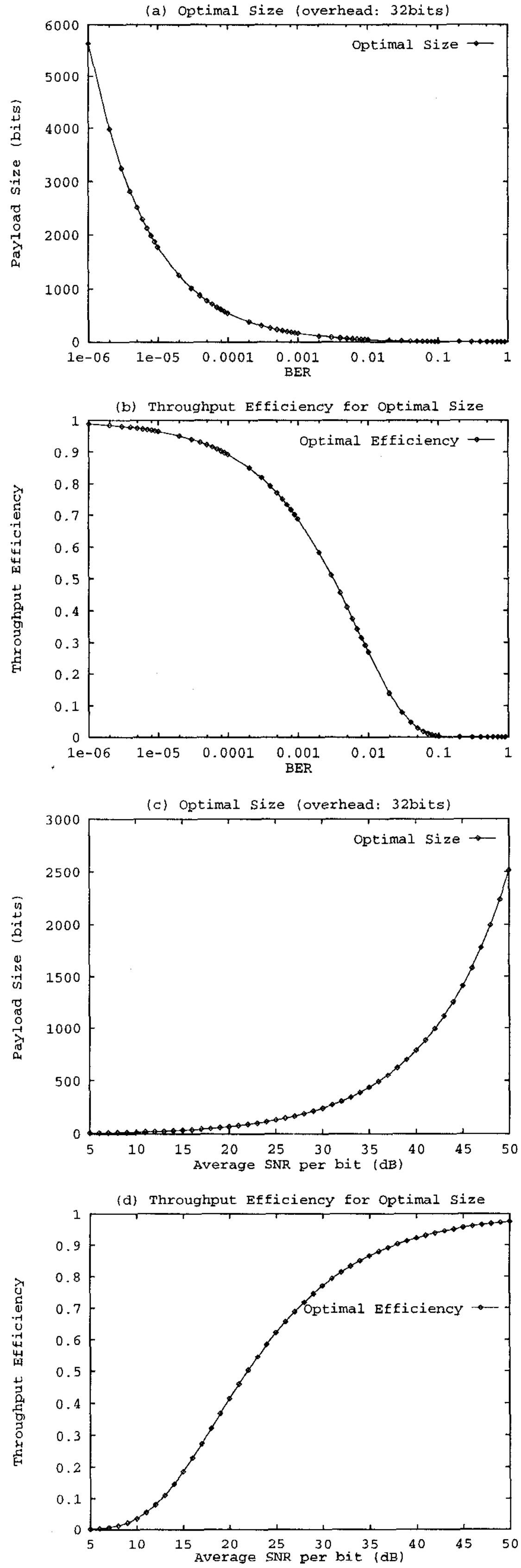

Figure 2: Optimal Packet Size and Throughput Efficiency of SDLP Protocol

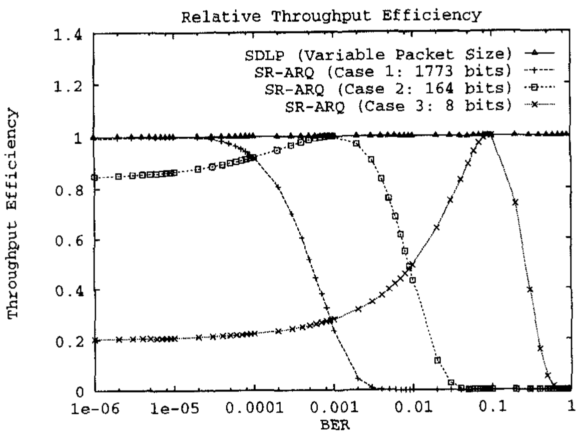

Figure 3: Comparison of SDLP and Classical SR-ARQ Protocols in terms of Relative Throughput Efficiency

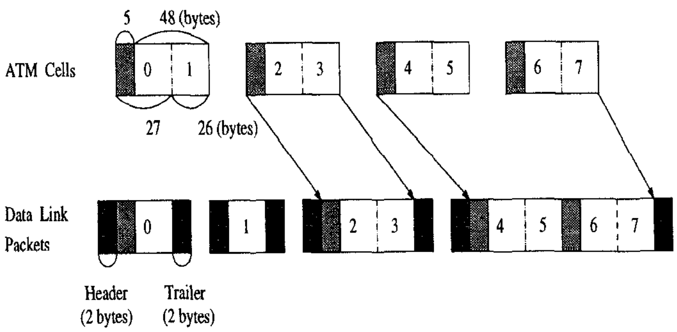

Figure 4: Packing Scheme for ATM Cells

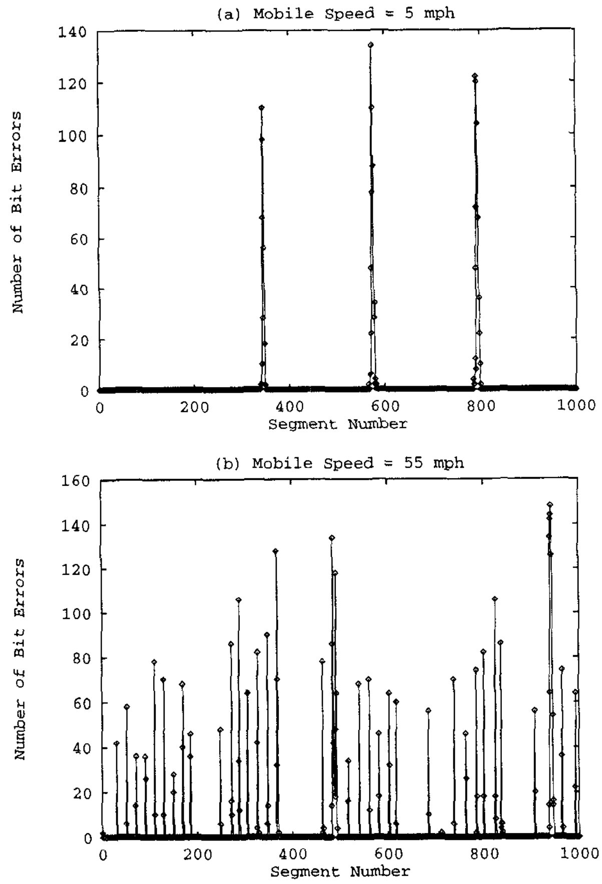

Figure 5: Error Distribution over Simulated Wireless Channel

as 0.27. Therefore, the segment size should be chosen before the curve of the throughput efficiency drops rapidly, so as to achieve the reasonable throughput efficiency. Since the optimal payload size is about 200 bits at the BER 10^{-3}, an ATM cell can be divided into 2 segments, 216 bits and 208 bits each. Figure 4 illustrates that a data link packet (the format is given in Figure 1(a)) includes as many segments as desired according to the channel condition at that time. The segment counter SC represents the information payload size of the data packet in terms of segments. At the receiver, SC is used to collect two consecutive segments for reassembly of an ATM cell. After that, each ATM cell is passed up to the ATM layer.

4 Experimental Results

We consider a simulation setup for wireless ATM. For simulation parameters, the carrier frequency is 2.4 GHz ISM band, the data rate is DS1 (1.544 Mbits/s), and mobile terminal speeds are 5 and 55 mph, respectively. Each mobile terminal has a wireless link with an ATM switch, which works as a base station. At the 2.4 GHz band and normal mobile speeds, the Doppler shift is limited up to 200 Hz. For example, the Doppler shift is about 18 Hz for pedestrians at 5 mph and 197 Hz for the mobile terminal speed of 55 mph. Since two source bits are mapped into one channel symbol at a time in $\pi/4$-DQPSK, the symbol interval is about 1.3 μsec at DS1 data rate.

Typical channel results are shown in Figure 5 for mobile terminal speeds 5 mph and 55 mph each. Number of bit errors for each segment is plotted against the segment number. Each ATM cell is divided into two segments, based on the packing scheme in Section 3.2. For pedestrians at 5 mph, there are a few bursts of segments in error, and in each burst the number of most bit errors per segment is large. Since the channel is usually error-free, it would be possible to use only an ARQ-based scheme without FEC (Forward Error Correction). On the other hand, for the mobile terminal speed of 55 mph many segments are in error. That is, 102 segments are in error out of 1000 segments. Since errors are dispersed over segments in this case, the appropriate solution is to use a hybrid scheme where FEC is combined with an ARQ-based scheme. For example, when a rate-1/2 convolutional code is used here, the remaining error pattern after error correction is similar to the former walking case. Therefore, a hybrid scheme can be justified as a good solution.

The implementation of the protocol was carried out in the user space with UNIX sockets. When the round trip delay is assumed about 100 msec for the worst case over mobile radio channels, the maximum window size is 64 and the buffer size is 128 Kbytes (i.e., $4 \times 64 \times 0.5K$ bytes). Given simulation parameters above, we made several measurements based on the implementation. Figure 6 shows the normalized response time as a function of the frequency of control packets m, provided that the best response time is one. Since there is a tradeoff between response time and bandwidth to be consumed by the control packets, the frequency of control packets should be determined to balance them. For quality critical traffic, the initial value m_0 can start with 2 or 3 in order to obtain a good response time (about 1.1) at the ex-

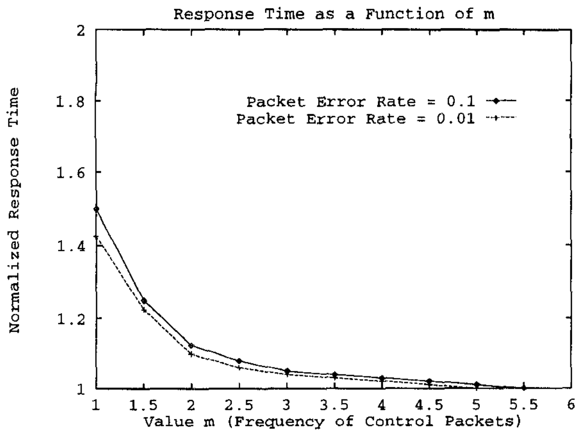

Figure 6: Response Time as a Function of m

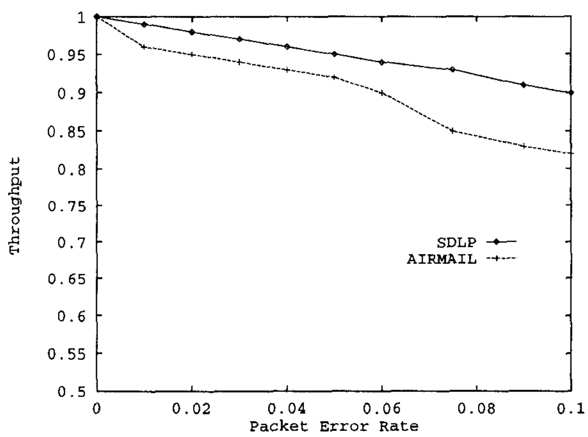

Figure 7: Comparison of SDLP and AIRMAIL in terms of Throughput

pense of small additional bandwidth (less than 5%). For time critical traffic, control packets could be sent more frequently to obtain a better response time. For example, if the receiver sends 5 control packets per round trip delay (i.e., $m_0 = 5$), the response time corresponds to the best case.

Finally, our protocol is compared with AIRMAIL, a latest protocol proposed by Lucent technologies for wireless networks in [1], in terms of throughput and response time. Since AIRMAIL is also based on the periodic status message, the response time is almost the same for both cases. The throughput is measured as the number of correct data packets and the total number of data packets including retransmissions for each packet error rate. As shown in Figure 7, the SDLP protocol gives better throughput, especially as the packet error rate becomes higher. In particular, the throughput of SDLP is as high as 0.9 for a packet error rate of 0.1 in comparison with 0.82 of AIRMAIL [4].

5 Conclusions

In this paper, we have discussed the design and performance of a novel ARQ protocol for wireless ATM. The key ideas in the protocol design consist of variable packet size and periodic status message. The packet size varies with the channel conditions, maximizing the throughput efficiency. The status messages are sent periodically from the receiver to avoid redundant retransmissions by obsoleting the timeout mechanism. We have also presented experimental results from a software emulator based on the UNIX socket implementation. The results showed the response time and throughput of the protocol. Since the use of FEC with retransmission is particularly suitable for high-speed mobiles over impaired wireless links, an extension of this work will be to investigate hybrid techniques, attempting to find an optimal solution between FEC and ARQ for wireless ATM networks.

References

[1] E. Ayanoglu, S. Paul, T. F. LaPorta, K. Sabnani and R. D. Gitlin, "AIRMAIL – A Link Layer Protocol for Wireless Networks," *ACM Journal of Wireless Networks*, February 1995.

[2] S. Nanda, R. Ejzak and B. T. Doshi, "A Retransmission Scheme for Circuit-Mode Data on Wireless Links," *IEEE Journal of Selected Areas in Communications*, October 1994.

[3] A. N. Netravali, W. D. Roome and K. Sabnani, "Design and Implementation of a High-Speed Transport Protocol," *IEEE Transactions on Communications*, November 1990.

[4] S. Paul *et al*, "An Asymmetric Protocol for Digital Cellular Communications," *IEEE INFOCOM'95*, pp. 1053-1062, 1995.

[5] J. G. Proakis, "Digital Communication", *McGraw-Hill*, 2nd edition, 1989.

[6] M. Schwartz, "Telecommunication Networks: Protocols, Modeling and Analysis," *Addison-Wesley*, 1987.