Vol. 1, Issue 3, pp.1115-1118

Analyzing Trends in the Mean Resequencing Buffer Occupancy and the Mean Resequencing Delay

Uday Kumar.J

Abstract— Under the assumption that all channels have the same transmission rate but possibly different time-invariant error rates, we derive the probability generating function of the resequencing buffer occupancy and the probability mass function of the resequencing delay. Then, by assuming the Gilbert–Elliott model for each channel, we extend our analysis to time-varying channels. Through examples, we compute the probability mass functions of the resequencing buffer occupancy and the resequencing delay for time-invariant channels. We analyze trends in the mean resequencing buffer occupancy and the mean resequencing delay as functions of system parameters.

Index Terms: *resequencing, buffer occupancy, delay*

I.INTRODUCTION

In a modern high-speed wireless data network, however, multiple parallel channels between adjacent transmitter–receiver pairs are often created using advanced wireless communication technologies (e.g., orthogonal frequency division multiplexing (OFDM) systems and multiple-input–multiple-output (MIMO) systems) to increase the data transmission rate. Unlike packet transmission over a single channel, in a multichannel communication system, multiple packets are sent at a time, one packet per channel, and packet transmission errors can occur across every channel. To implement error control through retransmission of packets in a multichannel communication system, an ARQ(AUTOMATIC-REPEAT-REQUEST) protocol has been generalized to allow concurrent transmission of multiple packets.

Today Communication plays a major role in any impact. In order to communicate the main thing we need is Networks. It must be of less cost ,An analytical approach for analyzing the mean % packet delay and mean queue length at the transmitting terminal in % wireless packet networks using the selective repeat (SR) automatic % repeat request high speed and must be Secure. But todays major problem in network is it faces heavy network traffic and Conjunctions.

Our project deals with this problem. It also solves the major problem faced due to this traffic and cost effective. Actually we won't take care of size of data that we used to

send to the receiver we just attach the document and send it to our receiver. But causes lot of problems in the network .

It not only affect our side but also for other user who depends on that particular network.

We have concentrate more on the network side because once the network gets down everyone of the user will be troubled. So our project gives the some rules to the user regarding the size of data. The Administrator may dynamically be able to change the size of data regarding to the network traffic that's it.

II. MODEL

Automatic repeat request

Automatic Repeat request (ARQ), also known as Automatic Repeat Query, is an error-control method for data transmission that uses acknowledgements (messages sent by the receiver indicating that it has correctly received a data frame or packet) and timeouts (specified periods of time allowed to elapse before an acknowledgment is to be received) to achieve reliable data transmission over an unreliable service. If the sender does not receive an acknowledgment before the timeout, it usually re-transmits the frame/packet until the sender receives an acknowledgment or exceeds a predefined number of re-transmissions.

The types of ARQ protocols include

- Stop-and-wait ARQ
- Go-Back-N ARQ
- Selective Repeat ARQ

These protocols reside in the Data Link or Transport Layers of the OSI model.

Go-Back-N ARQ

Go-Back-N ARQ is a specific instance of the Automatic Repeat-request (ARQ) Protocol, in which the sending process continues to send a number of frames specified by a window size even without receiving an ACK packet from the receiver. It is a special case of the general sliding window protocol with the transmit window size of N and receive window size of 1.

The receiver process keeps track of the sequence number of the next frame it expects to receive, and sends that number with every ACK it sends. The receiver will ignore any frame that does not have the exact sequence number it expects – whether that frame is a "past" duplicate of a frame it has already ACK'ed [1] or whether that frame is a "future" frame past the last packet it is waiting for. Once the sender has sent all of the frames in its window, it will detect that all of the frames since the first lost frame are outstanding, and will go back to sequence number of the last ACK it received from the receiver process and fill its window starting with that frame and continue the process over again.

Go-Back-N ARQ is a more efficient use of a connection than Stop-and-wait ARQ, since unlike waiting for an acknowledgement for each packet, the connection is still being utilized as packets are being sent. In other words, during the time that would otherwise be spent waiting, more packets are being sent. However, this method also results in sending frames multiple times – if any frame was lost or damaged, or the ACK acknowledging them was lost or damaged, then that frame and all following frames in the window (even if they were received without error) will be re-sent. To avoid this, Selective Repeat ARQ can be used.

Selective Repeat ARQ

Selective Repeat ARQ / Selective Reject ARQ is a specific instance of the Automatic Repeat-reQuest (ARQ) Protocol. It may be used as a protocol for the delivery and acknowledgement of message units, or it may be used as a protocol for the delivery of subdivided message sub-units.

When used as the protocol for the delivery of messages, the sending process continues to send a number of frames specified by a window size even after a frame loss. Unlike Go-Back-N ARQ, the receiving process will continue to accept and acknowledge frames sent after an initial error; this is the general case of the sliding window protocol with both transmit and receive window sizes greater than

The receiver process keeps track of the sequence number of the earliest frame it has not received, and sends that number with every ACK it sends. If a frame from the window with the subsequent sender does not reach the receiver, the sender continues to send subsequent frames until it has emptied its window. The receiver continues to fill its receiving frames, reply in each time with an ACK containing tg he sequence number of the earliest missing frame. Once the sender has sent all the frames in its window, it re-sends the frame number given by the ACKs, and then continues where it left off.

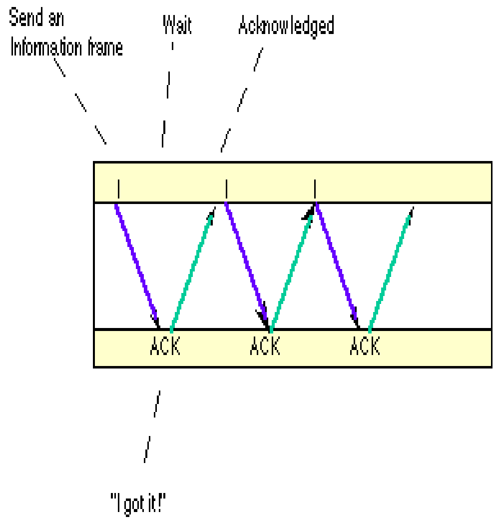
The size of the sending and receiving windows must be equal, and half the maximum sequence number (assuming that sequence numbers are numbered from 0 to $n-1$) to avoid miscommunication in all cases of packets being dropped. To understand this, consider the case when all ACKs are destroyed. If the receiving window is larger than half the maximum sequence number, some, possibly even all, of the packages that are resent after timeouts are duplicates that are not recognized as such. The sender moves its window for every packet that is acknowledged.

When used as the protocol for the delivery of subdivided messages it works somewhat differently. In non-continuous channels where messages may be variable in length, standard ARQ or Hybrid ARQ protocols may treat the message as a single unit. Alternately selective retransmission may be employed in conjunction with the basic ARQ mechanism where the message is first subdivided into sub-blocks (typically of fixed length) in a process called Packet segmentation. The original variable length message is thus represented as a concatenation of a variable number of sub-blocks. While in standard ARQ the message as a whole is either acknowledged (ACKed) or negatively acknowledged (NAKed), in ARQ with selective transmission the NAKed response would additionally carry a bit flag indicating the identity of each sub-block successfully received. In ARQ with selective retransmission of sub-divided messages each retransmission diminishes in length, needing to only contain the sub-blocks that were NAKed.

In most channel models with variable length messages, the probability of error-free reception diminishes in inverse proportion with increasing message length. In other words it's easier to receive a short message than a longer message. Therefore standard ARQ techniques involving variable length messages have increased difficulty delivering longer messages, as each repeat is the full length. Selective retransmission applied to variable length messages completely eliminates the difficulty in delivering longer messages, as successfully delivered sub-blocks are retained after each transmission, and the number of outstanding sub-blocks in following transmissions diminishes.

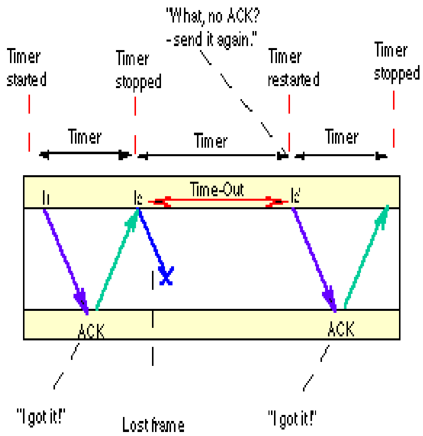
Stop and Wait ARQ

Stop and Wait transmission is the simplest reliability technique and is adequate for a very simple communications protocol. A stop and wait protocol transmits a Protocol Data Unit (PDU) of information and then waits for a response. The receiver receives each PDU and sends an Acknowledgement (ACK) PDU if a data PDU is received correctly, and a Negative Acknowledgement (NACK) PDU if the data was not received. In practice, the receiver may not be able to reliably identify whether a PDU has been received, and the transmitter will usually also need to implement a timer to recover from the condition where the receiver does not respond.
Under normal transmission the sender will receive an ACK for the data and then commence transmission of the next data block. For a long delay link, the sender may have to wait an appreciable time for this response. While it is waiting the sender is said to be in the "idle" state and is unable to send further data.



Stop and Wait ARQ - Waiting for Acknowledgment (ACK) from the remote node.

The blue arrows show the sequence of data PDUs being sent across the link from the sender (top to the receiver (bottom). A Stop and Wait protocol relies on two way transmission (full duplex or half duplex) to allow the receiver at the remote node to return PDUs acknowledging the successful transmission. The acknowledgements are shown in green in the diagram, and flow back to the original sender. A small processing delay may be introduced between reception of the last byte of a Data PDU and generation of the corresponding ACK.
When PDUs are lost, the receiver will not normally be able to identify the loss (most receivers will not receive anything, not even an indication that something has been corrupted). The transmitter must then rely upon a timer to detect the lack of a response.



III. C ONCLUSION

This paper mainly deals with the analysis of the resequencing buffer for SW-ARQ-inS over a generic number of parallel channels with both time-varying and time-invariant packet error rates. With the dynamic assignment rule applied in the protocol, exact statistical results of the resequencing buffer occupancy with both channel models were derived in steady state. The distribution function of the resequencing delay for the model with time-invariant error rates and the mean resequencing delay for the model with time-varying error rates were also obtained. For the model with time-invariant error rates,we numerically computed the pmf of the resequencing buffer occupancy using its probability generating function and the pmf of the resequencing delay. Through numerical and simulation results, we discussed the impact of the packet-to-channel assignment rules, the variance in the error states, the average error rate, and the number of parallel channels on the mean resequencing buffer occupancy and the mean resequencing delay.

IV. REFERENCES

[1] S. S. L. Chang, “Theory of information feedback systems,” IEEE Trans.Inf. Theory , vol. IT-2, pp. 29–40, 1956.

[2] M. E. Anagnostou and E. N. Protonotarios, “Performance analysis of the selective repeat ARQ protocol,” IEEE Trans. Commun. , vol.COM-34, pp. 127–135, 1986.

[3] M. Moeneclaey, H. Nruneel, I. Bruyland, and D. Y. Chung,“Throughput optimization for a generalized stop-and-wait ARQ scheme,” IEEE Trans. Commun. , vol. COM-34, pp. 205–207, 1986.

[4] Z. Rosberg and N. Shacham, “Resequencing delay and buffer occupancy under the selective-repeat ARQ,” IEEE Trans. Inf. Theory , vol.35, pp. 166–173, 1989.

[5] L. Badia, M. Rossi, and M. Zorzi, “SR ARQ packet pelay ptatistics on parkov channels in the presence of variable arrival rate,” IEEE Trans.Wireless Commun. , vol. 5, pp. 1639–1644, 2006.

[6] R. Fantacci, “Queueing analysis of the selective repeat automatic repeat request protocol wireless packet networks,” IEEE Trans. Veh. Technol. ,vol. 45, pp. 258–264, 1996.

[7] J. G. Kim and M. K. Krunz, “Delay analysis of selective repeat ARQ for a M arkovian source over a wireless channel,” IEEE Trans. Veh.Technol. , vol. 49, pp. 1968–1981, 2000.