Volume 2, No. 5, Sept-Oct 2011
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Analysis for ARQ Protocols on Multi Channels by using MIMD Congestion Control Algorithm

G.Sireesha*[1], C.Balarengadurai[2] and J.Srikanth[2]
[1]M.Tech student, [2]Associate Professor,
Aurora's Engineering College, Hyderabad, India
[1]sirisha.peas@gmail.com,[2]balamtech@yahoo.co.in,[3]jsrikanth@aurora.ac.in

Abstract: In this paper we proposed to improve channels transmission rate but different time-invariant error rates. By assuming the Gilbert–Elliott model (GEM)for each channel and TCP for high speed the additive increase multiplicative decrease algorithm used in the standard TCP, Scalable TCP uses a multiplicative increase, multiplicative decrease (MIMD) algorithm for the window size evolution , we extend our analysis to time-varying channels. We compute the probability mass functions of the sequencing buffer occupancy and the sequencing delay for time-invariant channels. Our approach is based on the logarithm of the window size evolution has the same behavior as the workload process in a standard G/G/1 queue. The Laplace-Stieltjes transform of the equivalent queue is then shown to directly provide the throughput of the congestion control algorithm (CCA) and the higher moments of the window size.

Keywords: MIMD, GEM, TCP, CCA, PGF, PMF.

I. INTRODUCTION

The main performance characteristics of a data transmission system with ARQ error control, delay, queue length. Various protocols which have been proposed use, as a part or as a whole, the basic selective-repeat ARQ protocol. Their performance analysis has, been restricted to throughput characteristics.Two different methods for the queue length and delay analyses are presented. The system is modeled as a discrete time queue with infinite buffer storage. Transmission errors are considered to independent, and block arrivals may follow an arbitrary inter arrival distribution. The first method uses an exact Markov state model, which the theory of absorbing and Markov chains is applied, leads to a computational algorithm. The second method, which is based on a specific assumption, uses a substantially simpler stochastic model and results in equations which are easily solved by means of iterative computation. In the case of geometrically distributed inter arrival times, simple analytical formulas are extracted [1].

A communication network that regulates retransmissions of erroneous packets by a selective-repeat (SR) automatic repeat Request (ARQ) protocol. Packets are assigned consecutive integers, and the transmitter continuously transmits them in order until a negative acknowledgment or a time-out is observed. The receiver, upon receipt of a packet, checks for errors and returns positive/negative acknowledgment (ACK/NACK) accordingly. Only packets for which either NACK or time-out have been observed are retransmitted. Under SR ARQ, the receiver accepts packets that are out of order and must store them temporarily if it has to deliver them in sequence.

Tx and Rx are transmitter and receiver, respectively. The re-sequencing buffer requirements and the resulting packet delay constitute major factors in overall system considerations. The distributions of the buffer occupancy and the re-sequencing delay at the receiver under a heavy traffic situation. This enables the network designer to determine how much buffer capacity at the receiver will guarantee certain specified performance measures. The retransmission of erroneous packets depends on the particular ARQ protocol used. There are three basic ARQ schemes: stop-and-wait, go-back-N, and selective-repeat (SR).

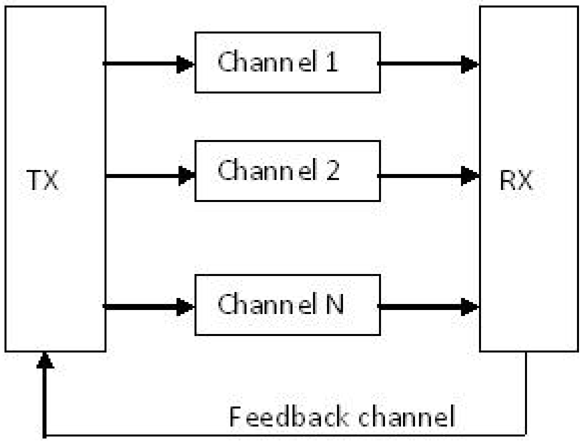

Figure 1: Basic Structure

In stop-and-wait, the transmitter remains idle after a packet's transmission until ACK, NACK, or a time-out is observed. In go-back-N protocol, packets that arrive out of order are ignored by the receiver which does not have to allocate any buffers to them. Under the selective-repeat ARQ packets must be stored in the receiver's buffers until they can be sent out in the original order.

The buffer needed for this purpose is referred to as a re-sequencing buffer, and the time that packets spend there as re-sequencing delay. Two types of physical systems protocols are: 1. A Slotted Time Channel, Constant Packet Length, and Heavy Traffic. 2. An Arbitrary Arrival Process and Variable Packet Length. Three types of methods are used they are: The Probability-Generating Function, The First Two Moments - Recursive Formulas, The PGF and the First Moment, The Packet re-sequencing Delay [2].

A generalized stop-and-wait ARQ scheme, data blocks are sent to the receiver in sets containing an arbitrary number of identical copies. The main idea behind this scheme is a reduction of the idle time of the transmitter. For small values it can be verified that the performances of the three schemes. For moderate large values, on the contrary both the conventional optimized scheme is outperformed by the optimum generalized scheme, whose throughput appears to be quite insensitive to the error prone [3].

Analyze the mean delay experienced a Markovian source over a wireless channel with time-varying error characteristics. The wireless link implements the selective-repeat automatic repeat request (ARQ) scheme for retransmission erroneous packets. We obtain good approximations of the total delay, which consists of transport and re-sequencing delays. The transport delay, in turn, consists of queuing and transmission delays. Our analysis accommodates both the inherent correlations between packet inter- arrival times. Traffic burstiness) and the time-varying nature of the channel error rate. The probability generating function (PGF) of the queue length under the “ideal” SR ARQ scheme is obtained and combined with the retransmission delay to obtain the mean transport delay. For the re-sequencing delay, the analysis is performed under the assumptions of heavy traffic and small window sizes (relative the channel sojourn times).The inaccuracy due to these assumptions is observed to be negligible. The autocorrelations in the arrival process or the time-varying nature of the channel state can lead to significant underestimation of the delay performance, particularly at high channel error rates [4].

An analytical framework for radio link level performance evaluation under scheduling and automatic repeat request (ARQ)-based error control in a multi-rate wireless network. The multi-rate transmission is assumed to be achieved through adaptive modulation and coding (AMC) in a correlated fading channel. The analytical framework, which is developed based on a vacation queuing model, can be applied to any scheduling scheme as long as the evolution of the joint service/vacation and channel processes can be determined. The exact statistics of queue length and delay are obtained and the radio link level throughput is calculated under both saturated and non-saturated buffer scenarios. The performance of max-rate (MR) scheduling scheme which exploits multiuser diversity and compare its performance with the round-robin (RR) scheduling scheme. The MR scheduling always results in higher throughput than the RR counterpart, RR scheduling offers better delay performance, MR scheme under light traffic load conditions. Applications for cross-layer design and packet-level admission control under delay constraints. This analytical framework would establish the base for fair comparison among different scheduling schemes and facilitate performance prediction at the higher layers in the protocol stack [5].

The packet delay statistics of a fully reliable Selective Repeat ARQ (SR ARQ) scheme is investigated. An N-State Discrete Time Markov Channel model is used to describe the packet error process and the channel round trip delay is considered to be non zero, i.e., ACK/NACK messages are received at the transmitter *m* channel slots after the packet transmission started. The ARQ packet delay statistics is evaluated by means of an exact analysis by jointly tracking packet errors and channel state evolution. To derive a Markov Channel description of a Rayleigh fading process are discussed and the delay statistics obtained from the Markov analysis is compared with that estimated by simulation of the SR ARQ protocol over the actual fading process. The accuracy of the delay statistics obtained from the Markov Channel representation of the actual fading process is investigated by explicitly addressing the effect of the number of states considered in the Markov channel model and the impact of the Doppler frequency. Finally, new analysis to obtain link layer statistics over N-State Markov channels on the adequacy of the widely used Markov modeling approach for the characterization of higher layer performance. But result of the delay analysis has been extended to the unreliable feedback case insights on the impact of ACK/NACK errors [6].

The behavior of the Stop and Wait and Go Back-N error detection and retransmission (ARQ) protocols in an environment characterized by non random errors, Analytic models are developed for the case that the error process is modeled as a Markovian process. These models may be used to predict performance measures such as expected queue length and expected delay for the case of the Stop and Wait protocol and the *Go* Back-N protocol when N is very large. The models can also be used to determine maximum throughputs for both protocols. The results of these models are compared with simulation results for the Selective Repeat ARQ protocol [7].

The SW and GBN retransmission protocols must be generalized when used in a multichannel communications system. The generalization takes the form of packet-to-channel assignment rules. The packet-to-channel assignment rule is derived and important special cases are identified. Simulation results were used to packet-to-channel assignment impacts channel utilization when the channels are different. The optimal assignment rule produces a channel utilization that is better than the channel utilization [8].

A communications system in multiple parallel channels are available to carry traffic from a transmitter to a receiver, and an extension of the selective-repeat automatic repeat request (SR-ARQ) protocol that dynamically assigns packets to channels for each (re)transmission is presented. Because of selective retransmission, packets arrive at the receiver out of order and must be stored in a re-sequencing buffer. A queuing model for the re-sequencing buffer is constructed. The generating function of the buffer occupancy and the packet delay distribution are derived [9].

II. THE MODEL

Consider the class of Multiplicative Increase and Multip licative Decrease (MIMD) congestion control algorithm where each ACK results in a window increment of $\alpha - 1 > 0$ and a loss event is responded with a reduction of window size by a fraction $1 - \beta < 1$. Scalable TCP can then be viewed as an instance from this class with $\alpha = 1.01$ and $\beta = 0.875$. This motivates us to study the window behavior of MIMD congestion control algorithms for the purpose of studying Scalable TCP. We focus on the analytical performance study of these algorithms, and, hence, of Scalable TCP, in the presence of random as well as congestion losses. we present three models based on different assumptions on the window size. Then we present a general analysis of these models. Our approach is based on showing that an invertible transformation applied to the window size process results in a process that has the same evolution as the total workload process in a standard G/G/1 queue. The Laplace-Stieltjes transform of the equivalent queuing process thus obtained provides the throughput of the connection as well as the higher moments of the window size of the given MIMD algorithm.

III. ANALYSIS

Consider the following discrete time stochastic recursive equation Wn+1 = max (AnWn,1)…. (1) The process, {Wn}, can be viewed as a sequence of observations of a continuous time process sampled at certain, not necessarily equal, time intervals. The sequence An 2 (0, 1) is assumed to be stationary and ergodic. Taking the logarithm of equation (3), we obtain

log[Wn+1] = max(log[An] + log[Wn], 0).

Using the substitutions Yn = log[Wn], and Un = log[An] in the above equation, we obtain

Yn+1 = max (Yn + Un , 0)…. (2)

We now make the following observation: The recursive equation (4) has the same form as the equation describing the workload process in a G/G/1 queue observed at, say, just after an arrival Un denotes the 5 difference between the service time of the nth customer and the inter arrival time between the nth and the (n + 1) th customer. Since the introduced transformation, log(·), is invertible, there is a one to one correspondence between the processes {Yn, n ¸ 0} and {Wn, n ¸ 0}. This observation allows us to study the stability of the window process {Wn, n ¸ 0} via that of {Yn, n ¸ 0}. Furthermore, the analogy with queuing theory of the process {Yn, n ¸ 0} allows us to obtain the steady state moments of Wn. Theorem 3.1 Assume that E [logA0] < 0. Then there exists a unique stationary ergodic process.

{W * n}, defined on the same probability space as {Wn}, that satisfies the recursion (1).Moreover, for any initial value W0 = w, there is a random time Tw, which is finite with probability 1, such that Wn = W * n for all n >= Tw. If E [logA0] > 0 then Wn tends to infinity w.p.1 for any initial value W0 = w. The log transformation allows us to obtain the moments of Wn in the stationary regime (i.e., moments of W*n) from the Laplace-Steiltjes Transform (LST) of Y*n in the stationary regime (i.e., LST of Y * n). The LST of Y * n is given by G(s) = E [e - $^{sY\ *n}$]…(4) which is defined for s 2 S, where S is the region of convergence of G(s). For a given integer k ¸ 0, the kth moment of W* n is obtained as follows

E[(W*n)k] = E[exp(kY *n)] = G(−k)….(5) where −k is assumed to belong to S. If −k 62 S then the corresponding moment is 1. Thus, all finite moments of W* n can be obtained from the LST of Y * n . A similar analysis can be done for the stochastic recursive equation

Wn+1 = min(AnWn,B)….(6) by making the transformation Yn = log[B]−log[Wn]. The moments of W*n can then be obtained from the LST of Y * n using the relation E[(W*n)k] = E[Bk exp(−kY *n)] = BkG(k)…..(7)All the moments of W*n are finite since G(s) is finite for s ¸ 0.The recursive equation for model (i), as given by (1), is similar to equation(3). Therefore, the analysis of this model can be done along the lines of the analysis of (3). Similarly, the analysis of models (ii) and (iii) can be done along the lines of the analysis of (6). We note that the analysis of model (iii) is similar to that of model (ii).

The equivalent queuing system of model (iii) can be obtained by deleting the idle periods of the equivalent queuing system of model (ii). The throughput of the MIMD algorithm, or the first moment of the window size, under different models, can be obtained from equation (5) and (7). These two assumptions allow us to use a discrete state space, S = {0, 1, 2, ...} for Yn. Thus, Yn can be modeled as a discrete state space Markov chain. The state Yn = i corresponds to Wn = Bl®i. The transition probabilities for this model are shown in Figure 2. Let Pn(j), j 2 S, be the probability of Yn being in state j at the end of the nth RTT.

The probability of being in state j at the end of the (n + 1)th RTT is given by

Pn+1(j) = (1 − p)Pn(j − 1) + pPn(j + k), j >=1 ,p Pk i=0 Pn(i), j = 0…..(8)

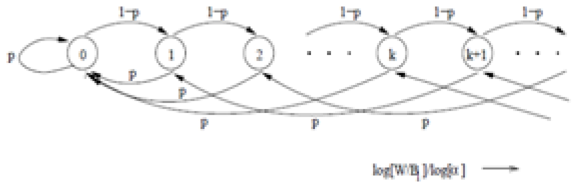

Figure 2: Transition probability of Y_n

Denote the z-transform of Yn by Yn(z). Yn(z) is defined as Yn(z) = ∑ Pn(j)zj.(9)

j=0

Upper Bound on Window Size and Window Dependent Random Losses: The probability of a loss in an RTT was

IV. EXPERIMENTAL EVALUATION

Upper Bound on Window Size and Window Dependent Random Losses: The probability of a loss in an RTT was independent of the window size in that RTT. In this section, we consider a model in which the losses in an RTT depend on the window size in that RTT. Specifically, we assume that each packet is dropped (or, equivalently, is in error) with a constant probability q. As a consequence of this assumption, the probability of packet drops in an RTT is no longer independent of the window size in that RTT. First, we present the model with window dependent losses. Then we propose an approximation to this model which will enable us to compute the throughput in the window dependent model using the expression for throughput in the window independent model (model (ii)). In each RTT, the window is reduced only once even in the presence of multiple packet drops. Loss recovery mechanisms of the recent TCP flavours such as New Reno and SACK. let Wn be the window size in the nth RTT. Let pn be the probability that the window is reduced in the nth RTT. Then, pn is given by pn = 1 − (1 − q)Wn…..(9) The window size evolution for this model can be written as Wn+1 = min(AnWn,Bu), where Bu is the upper bound on the window size, and An is now given by

An =α w.p. 1 − pn,¯β w.p. pn..…(10)

V. SIMULATION RESULT

Scalable TCP was proposed as a modification to the existing standard TCP for high speed networks. In the congestion avoidance phase, Scalable TCP uses the following algorithm to update the sender’s window at the end of every RTT: $W_{n+1} = 1.01 \times W_n$ if no losses are detected during the nth RTT, $W_{n+1} = 0.875 \times W_n$ if one or more losses are detected during the nth RTT. As mentioned in the Introduction, Scalable TCP is an instance of MIMD protocols, and therefore, we validate our models by performing simulations with Scalable TCP. The simulation are performed using Ns-2.The simulation setup has a source and a destination node.

The source node has infinite amount of data to send and uses Scalable TCP with New Reno flavor. The link bandwidth is 150 Mbps and the two way propagation delay is 120 ms. The window at the source is limited to 500 packets to emulate the receiver advertised window. The BDP for this system is approximately 2250 packets (packet size is 1040 bytes). In the Scalable TCP we have implemented in ns-2, the following assumptions are made: • the minimum window size, Bl, is 8. The growth rate of Scalable TCP is very small for small window sizes. It has been use the Scalable algorithm after a certain threshold. • There is no separate slow start phase since slow start can be viewed as a multiplicative increase algorithm with $\alpha = 2$. • For each positive ACK received, the window is increased by $\alpha - 1$ packets. When a loss is detected, the window is reduced by a factor of β. α is taken as 1.01 and β is taken as 0.86. This value of β gives $k = -\log[\beta]/ \log[\alpha] = 15$. We set α and β in this way so as to be close to the values recommended in ($\alpha = 1.01$, $\beta = 0.875$).

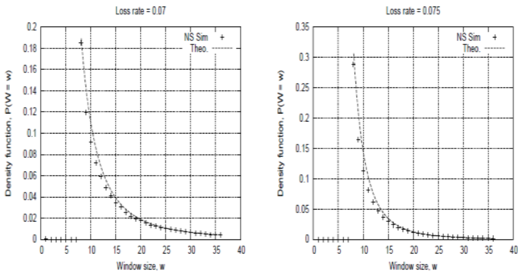

Figure: 3

$E[W_n] = 8n\ a/\ a - n$, respectively. In the simulations, the density function of W is obtained by sampling the window at an interval of RTT = 0.12s. We would like to note that the RTT is very close to the propagation delay in the present setting, and does not vary much. This results in a small discrepancy between the simulations and the theoretical function. The throughput in (TCP packets)*/RTT* as a function of the loss rate, *p*. The error bars are the 99% confidence intervals. Figure 8 shows the throughput in (TCP packets)*/RTT* as a function of the loss rate, *p*, for the model in which the maximum window at the sender is limited by the receiver's advertised window. The receiver buffer is assumed to be limited to 500 packets. The error bars are the 99% confidence intervals. A good match is observed between the simulations and the analysis two regions where model (i) and model (ii) are valid, respectively. As per approaches $1/(k + 1)$ from either direction, the approximate models (i) and (ii) diverge from the simulation results. However, model (i) gives a good estimate when $(k + 1)\ p >> 1$, i.e., $p >> 0.625$ ($k = 15$ in the simulations). Similarly, model (ii) gives a good approximation of the system when $p << 0.625$. The exact model fits well throughout the range of p. The throughput for model (i) is plotted for p ¸ 0.068 because a (in equation (18)) is > 1 for p>=0.0673.

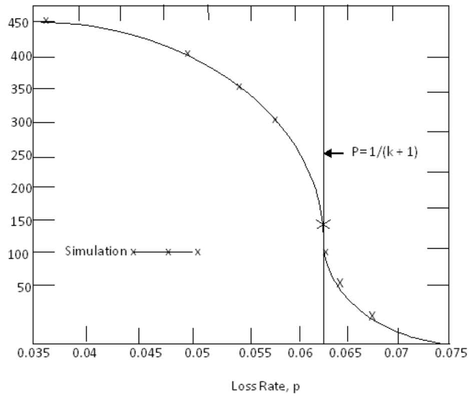

Figure 4: Throughput for maximum evaluation

VI. CONCLUSION

On ARQ protocol logarithm of the window size process of a connection using the MIMD congestion control algorithm is equivalent to the workload process in a G/G/1queue. The throughput of the connection and the higher moments of the window size process can be computed using the Laplace-Stieltjes transform of the equivalent workload process. For window independent losses, an exact expression can be obtained for the steady state probability distribution of the window size, and the throughput of the connection. In Future For window dependent losses an approximate expression, analogues to the square root formula for standard TCP, can be used to compute the throughput as well as SISD or MISD can be applied for calculating error rate for single and multiplicative channels when selective sequential queues are approached.

VII. ACKNOWLEDGEMENT

We would to thank the anonymous referee for helpful comments.

VIII. REFERENCES

[1]. 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.

[2]. 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.

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

[5]. L. B. Le, E. Hossain, and A. S. Alfa, “Delay statistics and throughput performance for multi-rate wireless networks under multiuser diversity,” IEEE Trans. Wireless Commun., vol. 5, pp. 3234–3243, 2006.

[6]. M. Rossi, L. Badia, and M. Zorzi, “SR ARQ delay statistics on N-state Markov channels with non-instantaneous

feedback," IEEE Trans.Wireless Commun., vol. 5, pp. 1526–1536, 2006.

[7]. D. Towsley, "A statistical analysis of ARQ protocols operating in a non independent error environment," IEEE Trans. Commun., vol. COM-29, pp. 971–981, 1981.

[8]. Z. Ding and M. Rice, "Throughput analysis of ARQ protocols for parallel multichannel communications," in Proc. IEEE GLOBECOM, 2005, pp. 1279–1283.

[9]. N. Shacham and B. C. Shin, "A selective-repeat-ARQ protocol for parallel channels and its resequencing analysis," IEEE Trans. Commun., vol. 40, pp. 773–782, 1992.

[10]. S. Floyd. HighSpeed TCP for Large Congestion Windows. RFC 3649, Experimental, December 2003. Available at www.icir.org/floyd/hstcp.html.

[11]. L. Xu, K. Harfoush, and I. Rhee. Binary Increase Congestion Control (BIC) for Fast Long-Distance Network. In Proceedings of the IEEE INFOCOM, March 2004.

[12]. F. Baccelli and D. Hong. AIMD, Fairness and Fractal Scaling of TCP Traffic. In Proceedings of the IEEE INFOCOM, April 2002.

Short Biodata of the Author

Ms. G.Sireesha received her B.Tech in Computer science and Engineering from Royal Institute of Technology and Science, JNTU, Hyderabad and Pursuing M.Tech in Computer science (Parallel computing) from Aurora's Engineering College, JNTU, Hyderabad.

