Throughput Analysis of TCP in Multi-Hop Wireless Networks with IEEE 802.11 MAC

Arzad A. Kherani *
INRIA,
2004 Route Des Lucioles-BP 93,
Sophia Antipolis,
France 06902.
Email: alam@sophia.inria.fr

Rajeev Shorey †
Computer Science Department,
National University of Singapore,
3 Science Drive 2,
Singapore 117543.
Email: rajeev@comp.nus.edu.sg

***Abstract*— We consider the problem of modeling TCP over multi-hop wireless networks using the IEEE 802.11 protocol. By identifying suitable regeneration instants, we are able to apply the standard technique of regenerative processes to compute the long term average throughput achieved by a single TCP session. Simulation results show that the proposed model predicts the TCP throughput to a very high level of accuracy. We then discuss how to extend this model to more general situations.**

I. INTRODUCTION

Ease and speed of deployment of wireless ad hoc networks has made these networks being used increasingly for personal area networking (e.g., cell phone, laptop, ear phone), military environments and emergency operations (search-and-rescue, policing and fire fighting). IEEE 802.11b [1] has emerged as the most widely used Wireless LAN standard in commercial environments.

An important problem in such scenario is that of predicting the interaction between TCP and the 802.11 protocol [1]. There is significant amount of literature available on performance of TCP over ad hoc networks employing the 802.11 protocol, (see [10], [8], [5], [11], [12], and the references therein). Most of the earlier studies of TCP performance over multihop 802.11 networks are based on simulations where the performance measure of interest is the *goodput* obtained by a TCP controlled FTP application transferring a file of infinite length.

In [3], the authors examine the problems with IEEE 802.11 MAC protocol in multi-hop networks. They present two problems existing in IEEE 802.11 based multi-hop wireless ad hoc networks. These are the TCP instability problem and the unfairness problem. In [4], the authors study the interactions of the IEEE 802.11 MAC and ad hoc forwarding and the effect on capacity for several simple configurations and traffic patterns. The paper examines the capacity of wireless ad hoc networks via simulations and analysis from first principles. In both these papers, the authors perform a simulation study backed by very simple analytical arguments.

Our objective in this paper is to propose a mathematical model for TCP over IEEE 802.11 based wireless ad hoc networks. The modeling and analysis of TCP over IEEE 802.11 based ad hoc network is hard due to several reasons. First, the IEEE 802.11 MAC protocol [1] is a complex protocol that involves a four-way handshake for each transmission attempt. Second, the TCP protocol is characterized by an end-to-end closed loop flow control; in contrast, IEEE 802.11 MAC is a closed loop flow control on a per link basis. The interaction of TCP with IEEE 802.11 MAC thus becomes complex. In addition, wireless networks based on IEEE 802.11 suffer from what is known as the "hidden-node problem" [2]. To reduce collisions caused by hidden terminals in the network, 802.11 uses a four-way RTS/CTS/DATA/ACK exchange. The dynamics of the four way handshake (RTS/CTS/DATA/ACK) coupled with the closed-loop nature of TCP make the study of TCP over such networks a challenging task.

In this paper, we develop a general methodology for analysing TCP over IEEE 802.11 networks and study this with help of simple examples. In an another paper [9], we consider a more general methodology for modeling TCP over IEEE 802.11 networks and apply it to study the effect of "ACK thinning" due to the delayed acknowledgement option of TCP.

The paper is organised as follows. In Section II, we discuss the approach to TCP modeling over an end-to-end chain of nodes with the help of a simple example. Section III discusses methods of extending the analysis in the previous section to a more general scenario. In Section IV, we present the simulation results. This is followed by a brief discussion on the performance of TCP with delayed acknowledgements in Section V. Finally, Section VI concludes the paper.

II. AN APPROACH TO TCP MODELING OVER AN END-TO-END CHAIN

We now propose a general method for modeling TCP over a multi-hop IEEE 802.11 network. This study is meant to provide an understanding of the effect of various parameters on TCP performance. Unlike other related work, we also

* Work was supported by grant 2900 IT-1 from *Centre Franco-Indien pour la Promotion de la Recherche Avancee (CEFIPRA).*
† Rajeev Shorey is a Research Staff Member in IBM India Research Laboratory, New Delhi, India. He is currently on leave from IBM IRL.

allow for a file transfer of finite length and compute the expected TCP throughput as a function of the file transfer volume. In this section, for ease of presentation, we present the general methodology via a simple example under simplifying assumptions. This analysis gives a *closed form* expression for TCP throughput under the simplifying assumptions. Later in the paper, we remark on how to relax the assumptions made in this section.

A. Description of the Experimental Setup

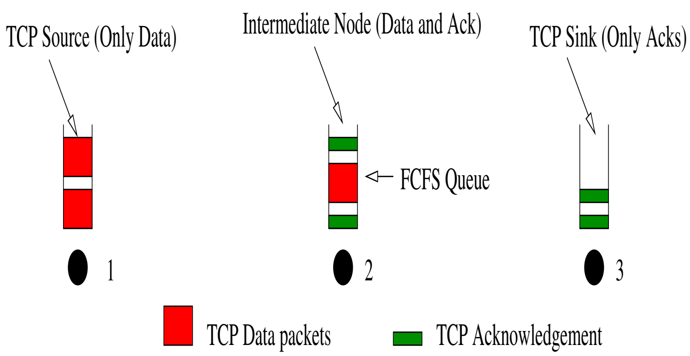

Fig. 1. A Linear Chain Network with 3 Nodes. The MAC Layer uses the 802.11 Protocol whose Transmit Buffers are also shown.

We begin by studying a linear chain of N nodes. A more general two-dimensional ad hoc network topology is very complex to analyse due to cross traffic from multiple sources and a greater interference from neighbouring nodes. Figure 1 depicts an 802.11 based linear chain network with three ($N = 3$) nodes numbered 1, 2 and 3. Node 2 is in the transmission range of both node 1 and node 3. Nodes 1 and 3 are not in each other's transmission range. For the time being assume that the transmission range and the sensing range (or the interference range) are equal. Node 1 is the source of a TCP session controlling a file transfer between node 1 and node 3. The size of the file being transmitted is V TCP packets. The nodes are assumed to be static; this, along with assumption of a routing protocol such as DSR [18], implies that the routing is static. The MAC layer implements the four way handshake (RTS/CTS/DATA/ACK) for data transmission ([1], [2], [3]). Further, for analytical tractability, we assume that TCP's window size is fixed and is equal to W packets.

Note that in the scenario under consideration, we can have atmost one successful data transmission going on in the network at any instant. The assumptions imposed on the system imply that we have relaxed two constraints of adhoc networks, (i) dynamic routing owing to mobility/failure of nodes and, (ii) multiple simultaneous transmissions in the network. We are interested in the effect of packet scheduling algorithm used at the MAC layer and the hidden node problem [2].

We assume that at node 2, the ACK and DATA packets are in two separate queues and are served probabilistically (as in Weighted Fair Queueing (WFQ) [15]). This is implemented as follows: node 2 maintains two queues, one for DATA packets and the other for ACK packets. After sending a packet (DATA or ACK) successfully and if the two queues are both nonempty, node 2 selects a packet from ACK queue for transmission with probability p_a. We will later show how to remove this assumption and incorporate FCFS queue at node 2 in our model. Note that nodes 1 and 3 can have only DATA and ACK packets respectively thus requiring only a FCFS queue.

We assume that the backoff values (in number of slots) are sampled from a Geometric distribution with mean $\frac{1}{p}$ instead of Uniform distribution (see [13], [19, Chapter 8]). Further, we assume that there is no exponential backoff, i.e., the mean of the successive (Geometrically distributed) backoff timer values are the same irrespective of the number of collisions. We do this in order to make the analysis more tractable. We later show that we actually need none of these two assumptions; however for now we assume that *both* of these hold.

B. The Embedded Markov Chain

We illustrate the general methodology used for modeling TCP over a multi-hop network of Figure 1 by considering a simple case where TCP's window size W is fixed at 2, thus requiring lesser notation while conveying the general idea.

For $n \geq 1$, let us introduce the following notation

- Z^n denote the time instant at which the n^{th} successful transmission is complete in the network, i.e., at instant Z^n the n^{th} MAC layer acknowledgement is received by one of the three nodes. We use the convention that $Z^1 = 0$.
- $P_i^{(n)}$ denote the number of packets in the transmit buffer of node $1 \leq i \leq 3$ at instant Z^n.
- $S^{(n)}$ denote the 2-dimensional vector corresponding to the transmit queue of node 2 at instant Z^n. $S^{(n)}(1) = p, a$ depending on whether the packet waiting to be transmitted from node 2 is a DATA packet or an ACK packet; we let $S^{(n)}(1) = 0$ is $P_2^{(n)} = 0$. Similarly, $S^{(n)}(2) = p, a, 0$ depending on the status of the next awaiting packet (if available) at node 2.

It follows that the process $\{P_1^{(n)}, S^{(n)}, P_3^{(n)}\}$ embedded at instants Z^n forms a Markov chain [16]. At instant Z^n, all the nodes that have packets to transmit start decreasing their residual backoff timers (since they have just heard a MAC layer ACK). The assumption of Geometrically distributed backoff timer values, along with the assumption that there is no exponential backoff mechanism for the successive backoff timers, then implies that these residual backoff timer values will again be Geometrically distributed with the same mean as the original. Thus it is as if all the nodes have just now started their first attempt of transmission. This is the reason that the process $\{P_1^{(n)}, S^{(n)}, P_3^{(n)}\}$ is Markovian.

Observe that exactly one of either TCP DATA or TCP ACK packet is successfully transmitted in the interval $(Z^n, Z^{n+1}]$. This is an important observation that will be used in the analysis to follow.

Let R^n denote a random variable which is 1 if an ACK packet has been transmitted from node 2 to node 1 in interval

$(Z^n, Z^{n+1}]$ and $R^n = 0$ otherwise. Now, the throughput achived by TCP source is $\frac{E[R]}{E[Z]}$ [16] where

$$E[R] = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} R^i$$

and

$$E[Z] = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} (Z^{i+1} - Z^i)$$

Thus we can use the standard Markov renewal reward approach to find the TCP throughput. Clearly, the TCP throughput thus obtained will be a function of p_a and the mean of the backoff value $\frac{1}{p}$.

From now onward, we drop the time index n from the notation. Figure 2 depicts the evolution of the Markov chain $\{P_1, S, P_3\}$. Here S is viewed as a column vector. The alphabets indicated in circles close to the states are used as a lable for the corresponding state.

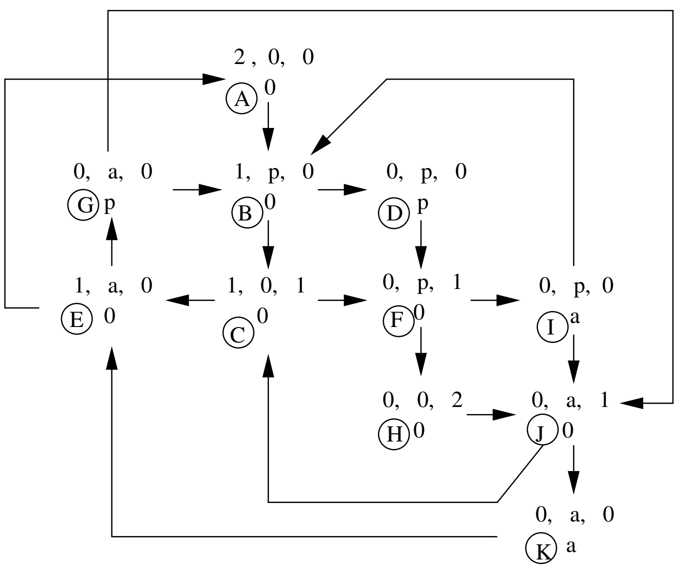

Fig. 2. Figure showing the Transition Diagram when $W = 2$.

The transition probability matrix of the transition diagram of Figure 2 is given in Table I. The notation used in Table I is as follows. Blank entries are zero, $p_{i,j}$ denotes the probability that node i's transmission becomes successful when nodes i and j both attempt to transmit. As mentioned before, p_a is the probabilistic priority given to the ACK packets over the DATA packets in node 2. Note from Figure 2 that there are no *self loops* for any state, this is because the chain is embedded at instants of completion of *successful* transmissions, Z^n.

Note that $p_{i,j} = 0.5$ for all $i \neq j$ if the distance between node 1 and 2 is same (or comparable) as that between node 3 and 2. For this case ($p_{i,j} = 0.5$ for all $i \neq j$) the transition probability matrix of Table I can be solved for the left eigen vector corresponding to the eigen value of unity thus giving us the stationary probability of this *embedded Markov chain* as a function of p_a. A simple calculation shows that the stationary probability of the embedded Markov chain $\{P_1^{(n)}, S^{(n)}, P_3^{(n)}\}$ is as indicated by the last column of Table I. *Note here that $\pi(C)$ is independent of p_a.*

C. TCP Throughput

In this section, we assume that the file to be transferred from the source to the destination is of infinite length. For finite length files, we need to perform numerical computation for a transient Markov chain. In the case of finite length files, there is an absorbing state (the state corresponding to the zero residual file) and hence the numerical scheme solves the Markov chain only till the absorbing state.

We recall the notation R^n and Z^n. Note that $R^n = 1$ only for the following transitions: $I \to B$, $J \to C$, $K \to E$, $E \to A$, $G \to B$; and $R = 0$ otherwise. Further, the mean time between two consecutive successful transmissions (assuming no collisions) is given by,

$$\begin{aligned} E[Z^{n+1} - Z^n] &= DIFS + T_b + T_R + 3SIFS \\ &\quad + T_C + T_D + T_M \end{aligned} \tag{1}$$

where T_b is the expected backoff timer for the transition and T_R, T_C, T_M, T_D are time required for transmission of RTS, CTS, MAC layer acknowledgement and Data respectively; the DATA could be TCP ACK or TCP DATA packet depending on the transition. DIFS and SIFS are standard parameters of 802.11 protocol [1]. Note that, under assumption of Geometrically distributed backoff timers and no exponential backoff, and with δ used to denote the MAC slot time,

$T_b = \frac{\delta}{p}$ for transitions from a state where a single node has both the packets, and

$T_b = \frac{\delta}{2p}$ for transitions from a state where two distinct nodes have packets to be transmitted (this is because the minimum of 2 Geometrically distributed random variable with mean $\frac{1}{p}$ is again a Geometrically random variable with mean $\frac{1}{2p}$).

Let l_p denote the TCP DATA packet length and l_a denote the TCP ACK packet length. Thus, T_D being the time taken to transmit the data, if the link capacities are C, then $T_D = \frac{l_p}{C}$ or $\frac{l_a}{C}$ depending on whether the transition is due to transmission of a TCP DATA packet or a TCP ACK packet.

It follows, after some simple algebra, that TCP throughput is given by

$$\frac{20}{80T_O + (34 + 2p_a)\frac{l_p}{C} + (46 - 2p_a)\frac{l_a}{C} + \frac{55\delta}{p}} \tag{2}$$

where $T_O = DIFS + 3SIFS + T_R + T_C + T_M$ is the MAC layer overhead (required for handshake) for transmission of a data packet.

Remark Note that this expression is not very sensitive to p_a. This is to be expected since a large priority to ACKs will delay the transmission of DATA packets from node 2 to node 3 thus reducing the rate of ACK packet generation. On the other hand, small priority to ACK packets at node 2 will slow the rate of ACKs reaching node 1.

	A	B	C	D	E	F	G	H	I	J	K	$\pi(\cdot)$
A		1										$\frac{3-p_a}{40}$
B			$1-p_{1,2}$	$p_{1,2}$								$\frac{3+4p_a}{40}$
C					$1-p_{1,3}$	$p_{1,3}$						$\frac{1}{8}$
D						1						$\frac{3+4p_a}{80}$
E	$1-p_{1,2}$						$p_{1,2}$					$\frac{3-p_a}{20}$
F								$p_{2,3}$	$1-p_{2,3}$			$\frac{2+p}{20}$
G		p_a								$1-p_a$		$\frac{3-p_a}{40}$
H										1		$\frac{2+p_a}{40}$
I		p_a								$1-p_a$		$\frac{2+p_a}{40}$
J			$p_{2,3}$								$1-p_{2,3}$	$\frac{7-4p_a}{40}$
K					1							$\frac{7-4p_a}{80}$

TABLE I

TABLE SHOWING THE TRANSITION PROBABILITY MATRIX OF THE EMBEDDED MARKOV CHAIN.

III. REMARKS ON THE GENERALITY OF THE MODEL

The following remarks argue the general validity of the model (and its analysis) presented so far.

(i) Only Geometric backoff timer distribution is required to ensure the Markovian nature of the process $\{P_1^{(n)}, S^{(n)}, P_3^{(n)}\}$, while exponential backoff can also be easily incorporated in the model by appending extra state corresponding to the backoff.

(ii) In the absence of exponential backoff, the assumption of Geometrically distributed backoff timer value is not required and a general (in particular, Uniform) backoff timer distribution can be used by appending extra states corresponding to the residual timer value. Thus, it is clear from this and the previous point that we need only one of the assumptions, either Geometrically distributed backoff timer or no exponential backoff.

(iii) The transmission rate of the links can be randomly varying owing to the changing surrounding conditions thus making the throughput dependent on the dynamics of the channel rate variation. Our model is easily modified to capture this effect as long as the channel state variation is Markovian (as is usually assumed in standard models for channel state variation [17]).

(iv) The assumption of WFQ at node 2 can also be relaxed to incorporate FCFS scheduling at node 2 by suitably appending extra states to the process $\{P_1^{(n)}, S^{(n)}, P_3^{(n)}\}$.

(v) The case of $W > 2$ is easily analysed with this method by suitably appending extra states to the process $\{P_1^{(n)}, S^{(n)}, P_3^{(n)}\}$. More specifically, now $S^n(1)$ and $S^n(2)$ will each represent $(W-1)$ dimensional vectors corresponding to the TCP DATA packets queue and TCP ACK packets queue at node 2 and another variable will be used to represent the type of packet waiting to be transmitted by node 2.

(vi) We can modify the transition probability matrix to account for other scheduling policies to study their comparative performance.

(vii) Equation 1 is based on the assumption that there is no collision at the MAC layer. This assumption can be removed in following ways:

 (a) by looking at the time instants embedded at events of either collision detection or successful transmission. This will require introducing additional states which will indicate whether the last transmission was successful or not,

 (b) or, by explicitly taking care of the possible collisions in Equation 1 itself.

(viii) In the paper, we have assumed that the transmission range and the sensing range of the nodes are same. In practice, however, the sensing range of a node is at least twice as much as the transmission range. This does not change the analysis and results of our model.

(ix) Note that our model benefits from the fact that there can be only one successful transmission in the network at a time. Thus, any such network (with more number of nodes) where the topology is such that there can be at most one successful transmission going on at any instant, the above analysis/model is applicable.

(x) The above analysis assumes that the TCP window size is fixed. The exact dynamics of the TCP protocol can be modelled by using a scheme as in [14] and references therein. In that case, our model will provide the model of [14] with the mean sojourn time in the $\cdot$/G/∞ queues and the routing probabilities required by their model.

IV. SIMULATION RESULTS

To validate our model, we have used the *ns* simulator [6] with the CMU wireless extensions [7] whose parameters are tuned to model Lucent Wavelan card at 2 Mbps data rate. The parameters of IEEE 802.11 standard have the following values. DIFS is equal to 50 μs, SIFS is equal to 10 μs, T_R is 352 μs and T_C is equal to 304 μs. [1].

The simulated scenario is as shown in Figure 1. It consists of three nodes with distances chosen so that there can be only one transmission in the network and that node 1 and 3 are not each other's neighbours so that the TCP packets from node 1 to node 3 go through node 2. In order to avoid routing overheads, we have used the DSR routing protocol [18] and

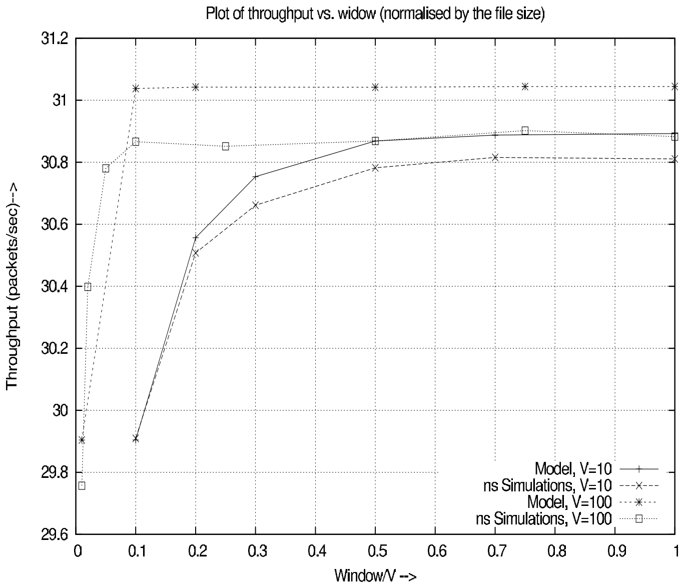

Fig. 3. Plot showing TCP Throughput for Files of Size 10 TCP Packets and 100 TCP Packets versus the TCP Window Size (Normalised by the File Size V). The Throughput Value is obtained from *ns* Simulations and Numerical Computation of the Analysis.

used a dummy TCP transfer between node 1 and 3 before starting the actual TCP transfer (whose average throughput is to be computed). This is repeated multiple times to get a reliable estimate of the average throughput.

Figure 3 plots TCP throughput for files of size V equal to 10 TCP packets and V equal to 100 TCP packets. The horizontal axis represents the simulated TCP window size normalised by the file size V. The value of throughput corresponding to the curves labelled “simulation” were obtained from *ns* simulations as mentioned above. Also shown in the plot are the value of throughput obtained from numerical computation of the analysis. Note that we realized all the possible TCP window sizes (starting from a window size of 1 packet till the file size). Since we also use window sizes (W) other than 2, Equation 2 does not hold and we have to compute the average throughput numerically. The numerical computation of average throughput from the model is very fast compared to the time required for repeated simulations for averaging. The simulation results shown in Figure 3 are for $p_a = 0.5$. We observed that using different values of p_a does not change the throughput much. It is seen from the plot that the model predicts TCP throughput very well.

V. Performance with Delayed Acknowedgements

We now study the effect of employing the delayed ACK feature of TCP client. We assume that the TCP's window size W is fixed at 2 and that the TCP client at node 3 generates an acknowledgement for every second TCP data packet it receives. A detailed study on the performance improvement of TCP with the delayed acknowledgement option is given in [9], [10].

Figure 4 depicts the evolution structure of the embedded

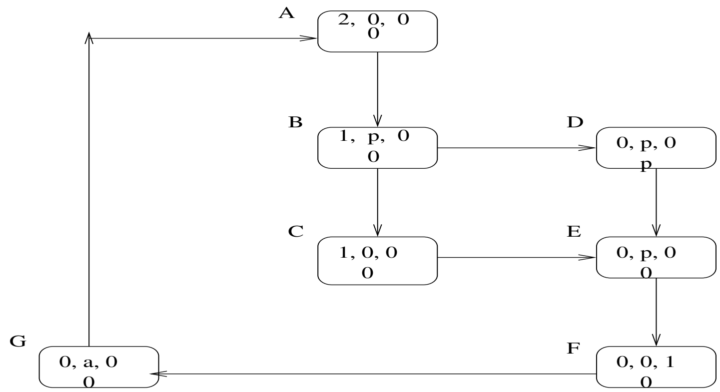

Fig. 4. Figure showing the Evolution Structure of the Embedded Markov Chain. The Transitions $B \to D$ and $B \to C$ are each with Probability $\frac{1}{2}$ and rest all the Transitions are with Probability 1.

Markov chain under the above assumption of TCP client employing the delayed acknowledgement feature. The transitions $B \to D$ and $B \to C$ are each with probability $\frac{1}{2}$ and rest all the transitions are with probability 1.

It is easily shown that $\pi(C) = \pi(D) = \frac{1}{12}$ and $\pi(\cdot) = \frac{1}{6}$ for rest of the states. The throughput achieved by TCP under delayed acknowedgement is then

$$T = \frac{E[R]}{E[Z]} = \frac{2}{6T_o + 4\frac{l_p}{c} + 2\frac{l_a}{c} + \frac{11\delta}{2p}} \tag{3}$$

Note that there is no p_a term in the above expression. Also note that the throughput achieved by TCP with delayed acknowledgement is higher than the case with no delayed ack.

It is to be noted that the Markov chains studied so far in the paper are periodic. The analysis presented so far is still valid since periodicity does not affect time averages, it only affects convergence in distribution. Therefore, we use $\pi(\cdot)$ just as we would without periodicity. The proof of the Markov Renewal Reward theorem is by looking at regeneration points at Markov renewal instants. Say we take visits to state i. Now the mean number of times the MRP visits state j is still given by the same expression as would have been had the chain been aperiodic. The calculation now follows.

VI. Conclusion

We have presented a general methodology for mathematical modeling of TCP over IEEE 802.11 networks using a simple topology; that of a linear chain of nodes. Simulations show that the model predicts the TCP throughput to a very high level of accuracy. We have also described how to extend this model to more general situations.

The model proposed in this paper has certain limitations. In [9], we extend the model proposed in this paper to eliminate some of its limitations. As part of the future work, it will be interesting to incorporate routing protocols (e.g., DSR, DSDV, AODV) in our model. Our model assumes that there can be at most one successful transmission going on in the network at any instant of time; it will be interesting to relax this limitation thus allowing for more number of nodes and multiple simultaneous successful transmissions.

We believe that the paper is an important step towards understanding the interaction between TCP protocol and multi-hop wireless ad hoc networks where the nodes are based upon the IEEE 802.11 MAC.

REFERENCES

[1] IEEE Computer Society LAN MAN Standards Committee. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.* New York, 1997. IEEE Std. 802.11-1997.

[2] V. Bharghavan, A. Demers, S. Shenker, and L, Zhang. MACAW: A Media Access Protocol for Wireless LANs. In Proc. ACM SIGCOMM Conference (SIGCOMM'94), August 1994.

[3] S. Xu and T. Saadawi, "Does the IEEE 802.11 MAC Protocol work well in Multihop Wireless Ad Hoc Networks", IEEE Communication Magazine, Vol 39, June 2001.

[4] J. Li, C. Blake, Douglas S. J. De Couto, Hu Imm Lee and R. Morris, "Capacity of Ad Hoc Wireless Networks", In Proceedings of the Seventh Annual International Conference on Mobile Computing and Networking (MOBICOM'2001), July 16-21, Rome, Italy, 2001.

[5] G. Holland and N. Vaidya, 'Analysis of TCP Performance over Mobile Ad Hoc Networks', *ACM Mobicom*, Seattle, Washington, 1999.

[6] K. Fall and K. Varadhan, *ns* Notes and Documentation. Technical report, UC Berkeley, LBL, USC/ISI, and Xerox PARC, November 1997.

[7] CMU Monarch Group. CMU Monarch extensions to *ns*. http://www.monarch.cs.cmu.edu/.

[8] A. Ahuja, S. Agarwal, J. P. Singh and R. Shorey, 'Performance of TCP over Different Routing Protocols in Mobile Ad-Hoc Networks', *IEEE VTC*, Tokyo, Japan, 2000.

[9] A. Kherani and R. Shorey, "Performance Improvement of TCP with Delayed ACKs in IEEE 802.11 Wireless LANs", in *IEEE WCNC 2004*, Atlanta, USA, March 2004.

[10] E. Altman and T. Jimenez, 'Improving TCP over multihop networks using delayed ACK', extended abstract, *MADNET'03*, Sophia-Antipolis, March 2003.

[11] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang and M. Gerla, 'The Impact of Multihop Wireless Channel on TCP Throughput and Loss', *IEEE INFOCOM'03*, San Francisco , March 2003.

[12] V. Ramarathinam and M. A. Labrador,'Performance Analysis of TCP over Static Ad Hoc Networks'. *15th International Conference on Parallel and Distributed Computing Systems (PDCS 2002)*, September 2002.

[13] F. Cali, M. Conti and E. Gregori, 'Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit', *IEEE/ACM Transactions on Networking*, Volume 8, Issue 6, Dec. 2000

[14] M. Garetto, R. Lo Cigno, M. Meo and M. A. Marsan; 'A Detailed and Accurate Closed Queueing Network Model of Many Interacting TCP Flows', *Infocom 2001*, Anchorage, Alaska, USA, April 22-26, 2001

[15] A. Demers, S. Keshav and S. Shenker, "Design and Analysis of a Fair Queueing Algorithm". In Proceedings of ACM SIGCOMM'89, Austin, September 1989.

[16] R. W. Wolff, *Stochastic Modeling and the Theory of Queues*, Prentice Hall, 1989.

[17] A. Kumar, "Comparative Performance Analysis of Versions of TCP in a Local Network with a Lossy Link", IEEE/ACM Transactions on Networking, August, 1998.

[18] D. Johnson, D. Maltz and J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi Hop Wireless Ad Hoc Networks", *In Ad Hoc Networking*, Edited by Charles E. Perkins, Addision-Wesley, 2001.

[19] A. Kumar, D. Manjunath and J. Kuri, "Communication Networking: An Analytical Approach", *To be published by Morgan Kaufman Publishers, San Francisco, USA*.