Modeling TCP over Ad hoc Wireless Networks using Multi-dimensional Markov Chains†

Shyamnath Gollakota, B. Venkata Ramana, and C. Siva Ram Murthy*
Department of Computer Science and Engineering
Indian Institute of Technology Madras 600036 India
{shyam,vramana}@cse.iitm.ernet.in and murthy@iitm.ac.in

Abstract— **The performance of Transmission Control Protocol (TCP) over Ad hoc wireless networks (or simply ad hoc networks) has been extensively studied through simulations by the research community. Although many theoretical models, such as [1], have been proposed for estimating the performance of TCP over wired networks, researchers have faced many difficulties in modeling TCP over ad hoc networks. These difficulties are mainly due to the behavior of the underlying physical and MAC layers. Recently, [2] attempted to solve this problem by simplifying the behavior of TCP, besides assuming that no packet losses occur. In this work, we attempt to provide a theoretical model for TCP by considering the main phases of TCP, namely the slow start phase and the congestion avoidance phase, thus providing a more accurate model that captures all of its main features. To the best of our knowledge, ours is the first model that considers the slow start phase while analyzing TCP's performance in ad hoc networks. We make use of multi-dimensional Markovian chains to model each of these phases. We then use the resulting steady state probabilities to estimate the goodput. Furthermore, the analysis is validated by comparing the theoretical and simulation results using various error models.**

I. Introduction

The Internet has experienced enormous growth over the past decade and is now positioned to provide a wide range of services, such as remote file access, digital libraries, and videoconferencing [3]. This has led to the emergence of Transmission Control Protocol (TCP) as the globally accepted Inter-networking solution for providing reliable communication over computer networks. Alongside, the recent performance improvements in computer and wireless technologies, have seen advanced mobile wireless communication replace (complement) traditional wired networks. Thus, there is an immediate need to study the behavior of traditional transport protocols like TCP over these advanced networks. Since ad hoc wireless networks are one of the most commonly found in many applications, we limit our discussion to these type of networks in what follows.

Ad hoc wireless networks are formed dynamically by a group of mobile nodes, without any fixed infrastructure and centralized administration. In these networks all the nodes are potentially routers, forwarding traffic on behalf of other users. These networks have proved to be very useful in on-the-fly conferences, electronic classrooms, and trade-shows. In these applications, where an infrastructure is unavailable or to deploy one is not cost effective, ad hoc networks come to the rescue by providing a readily deployable wireless network. The use of IEEE 802.11 with RTS/CTS makes the behavior of the protocols run on these networks considerably different from their wired counterparts. In addition, several problems arise due to shared nature of the wireless medium, limited transmission power (range) of wireless devices, packet losses, node mobility, and battery limitations. In this work, we deal with the effect of the first three on the performance of TCP over ad hoc networks.

TCP [4] is a connection-oriented, reliable, full-duplex transport layer protocol that is based on the end-to-end semantic. The main responsibilities of TCP are congestion control, flow control, in-order delivery, and reliable transportation of packets. TCP regulates its packet transmission by expanding and shrinking its *congestion window*. It uses an additive-increase and multiplicative-decrease strategy for changing the size of its congestion window as a function of the network conditions to handle the congestion situation in the network. TCP operates in two phases: *slow start* and *congestion avoidance*. In the slow start phase, starting from one packet, the window is increased exponentially by one packet for every acknowledgment (ACK) until the source estimate of the network capacity (also called the slow start threshold, *SSThresh*), is reached. Once the congestion window exceeds the *SSThresh*, TCP enters into the *congestion avoidance* phase, in which for every Round Trip Time (RTT) the size of the congestion window is increased by one packet.

The receiver sends a duplicate acknowledgment (DUPACK) for every out-of-order packet it receives. Two mechanisms are available for the detection of losses. First, on reception of three consecutive DUPACKs, interpreting it as a packet loss, the TCP halves the current congestion window and re-transmits the lost packet. Second, when no ACK is received before the re-transmission timeout, TCP updates the *SSThresh* to half of the current congestion window and resets the congestion window to one packet, (i.e., to one Maximum Segment Size, MSS) and goes to the slow start phase.

In this work, we attempt to provide a theoretical model for TCP by considering both the slow start and congestion avoidance phases, thus providing a more accurate model that captures all of its main features. To the best of our knowledge, ours is the first model that considers the slow start phase while

†This work was supported by the Department of Science and Technology, New Delhi, India.
*Author for correspondence.

1-4244-0425-8/06/$20.00 ©2006 IEEE

analyzing TCP's performance in ad hoc networks. We make use of multi-dimensional Markovian chains to model each of these phases while considering the effect of losses and MAC contentions. We then use the resulting steady state probabilities to determine the goodput. Analysis considering various other enhancements to TCP can be made on similar lines.

A. Prior Work

The performance of TCP is a widely studied problem in the research community. Many papers exist on the theoretical analysis of the performance of TCP over wired networks. [1] presents a simple model of the TCP protocol which captures the congestion avoidance behavior of TCP but does not consider the effect of the slow start phase. Also since the model is for wired networks, it does not capture the shared nature of the wireless links. Again, one can find a large number of papers, [5-7], on the measured performance of TCP over ad hoc networks. However, very little work has been done on the theoretical front. [8] makes some simplifying assumptions such as constant congestion window size and instantaneous ACK delivery in order to get an upper bound on the throughput of TCP over a IEEE 802.11 multi-hop string topology. [9] provides a more accurate Markovian model with the constant congestion window size assumption but for a two hop network. More recently, [2] extends the analysis to a general n hop network with the same assumptions. In addition, no packet losses are considered in any of the above papers. While considering the effect of packet losses on the performance, approximating TCP to a fixed congestion window size transmission does not yield the results required. Hence, we propose a modeling which considers TCP in its full form where the congestion window size changes depending on which phase the system is in.

B. Organization of the Paper

In Section II, we present a multi-dimensional Markovian model for TCP which considers the behavior of the MAC and the effect of losses. Section III explains the goodput calculation using the proposed Markovian model. In Section IV, we explain how the parameters used in the model are determined. Section V verifies the theoretical analysis by corroborating it with simulation results. Finally, Section VI concludes our work and provides future directions for research.

II. Markovian Modeling of TCP

In this section we provide the multi-dimensional Markovian model which captures the behavior of TCP over ad hoc wireless networks. We provide the system model and the assumptions that are considered while modeling. We analyze each of the phases of TCP and model them separately and provide the transitions between these phases.

A. System Model and Assumptions

Our system consists of a single TCP flow running over an ad hoc wireless network. We assume that the source has infinite data to send, and so the size of TCP packets is always equal to one MSS. The TCP packet service rate, U_t, is the typical number of TCP packets the system serves in a unit time when it is constantly busy. The ACK packet service rate, U_a, is the typical number of ACK packets the system serves in unit time when it is constantly busy [2]. The time unit being considered includes all the delays at the nodes along the path. Since the service rates are inversely proportional to the size of the packet and that of the ACK, the relation between U_t and U_a is $\frac{U_t}{U_a} = \frac{size\ of\ ACK}{size\ of\ packet}$. Thus, we limit our attention to calculate U_t in what follows since U_a can be derived from the above relation. Let us consider a very small interval of time, δ, during which an event can occur. An event can be either an ACK being transmitted by the receiver or a packet being transmitted by the sender. Thus, the probability that the packet (ACK) is generated in the small interval δ is given by δU_t (δU_a). In addition, in order to bring in the channel contentions, link access probabilities are introduced. Let p (q) denote the probability that the sender (receiver) captures the channel given that both the sender and receiver compete for it. Hence, when both are competing for the channel, the probability that the sender (receiver) succeeds in the small interval δ is given by $p\delta U_t$ ($q\delta U_a$). For simplicity, it is assumed that both the sender and the receiver compete for the channel with equal probabilities (i.e., $p = q$). Let us denote W_{Max} to be the maximum possible congestion window size. This constraint can be due to the limitations on the buffer size of either the sender or the receiver. We also assume that every node in the network under consideration has the same transmission range, carrier sense range, and interference range. Further, we exclude the possibility of out-of-order packet (ACK) delivery.

B. Markov Chains

A Markov chain $\{\mathbf{x}_n\}$ is a sequence of random values, $\mathbf{x}_n$, whose value at a time interval, n, depends upon the value at the previous time. The value of the Markov variable at the present time is called its state. The state of a Markov process is a random variable. Markov chains can be seen as a finite or countably infinite number of states $e_1, e_2, \cdots, e_i, \cdots$, such that the future evolution of the process, once it is in a given state depends only on the present state [10]. Thus, the probability of transition from state e_i to e_j is fixed and does not depend on the path by which the process arrived to state e_i. The matrix containing the transition probabilities from every state e_i to every other state e_j is called the transition probability matrix. An aperiodic, irreducible Markov chain has a steady state distribution, whose steady state probabilities are calculated from the global balance equations [11]. Let p_i be the steady state probability of the system being in state e_i. Let P_{ij} be the probability of transition from state e_i to e_j. Then we have $\sum_i p_i P_{ij} = p_j$ for all values of j. Let p be the row vector of the steady state probabilities and T be the transition probability matrix. Then the above equation can be expressed in the matrix form as $pT = p$.

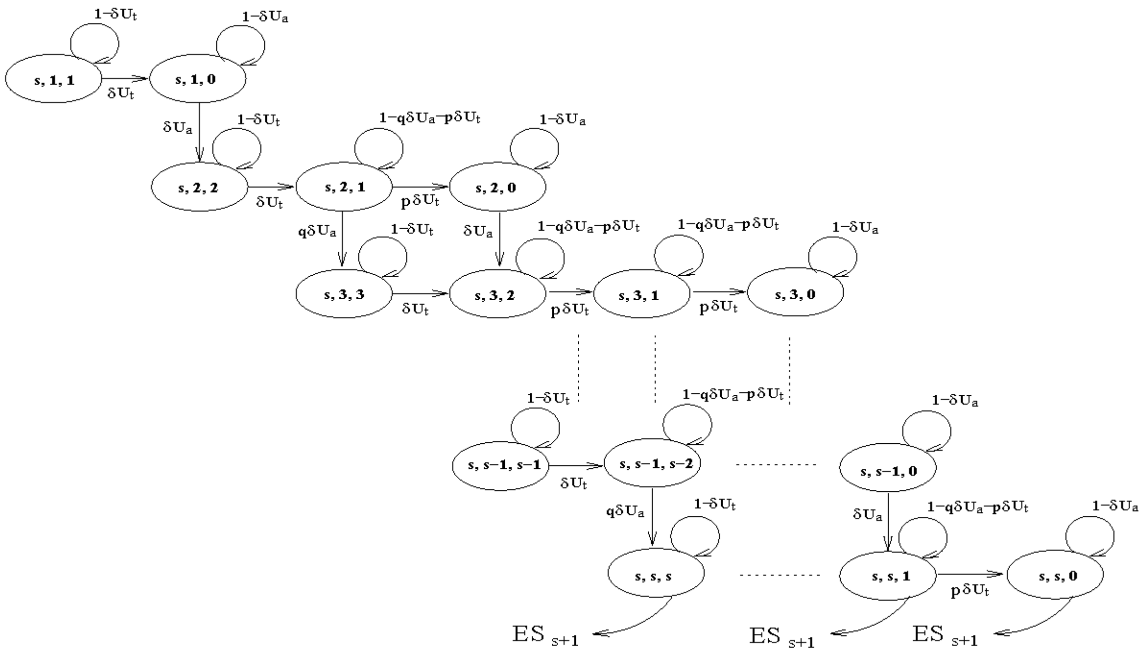

Fig. 1. Slow start phase (SS_s) with the value of $SSThresh$ being s, for various possible values of w and l. Here, ES_{s+1} denotes the Errorless round with congestion window size $s+1$.

C. Modeling Slow Start Phase

In the slow start phase, for every packet that is acknowledged, the congestion window size is incremented by one. However, when the current window size exceeds the $SSThresh$, TCP enters the congestion avoidance phase. In our model, we assume that no packet is lost in the slow start phase. This assumption is justified by the fact that when ever the system enters the slow start phase from the congestion avoidance phase, the $SSThresh$ is made half of the current congestion window size. So the $SSThresh$ can be considered to be less than the capacity of the network. This phase is modeled using a set of three-dimensional Markov states, $[s, w, l]$. We say that the system is in state $[s, w, l]$ if TCP is in the slow start phase, where

- s is the current $SSThresh$, $2 \leq s \leq \lfloor \frac{W_{Max}}{2} \rfloor$
- w is the current congestion window size, $1 \leq w \leq s$
- l is the number of packets left in the congestion window to be sent, $0 \leq l \leq w$.

As depicted in Figure 1, when the system is in states of the form $[s, w, 0]$ the sender has no packet to send, hence there is no competition for the channel from the sender. Thus, the only event possible is the ACK transmission by the receiver, which occurs with a probability of δU_a. Similarly, when the system is in state $[s, w, w]$, the receiver has no ACKs to send. Hence, there is no competition from the receiver and the only possible event is a packet transmission by the sender, which occurs with a probability of δU_t. In every other state of the form $[s, w, l]$, there is a competition between the sender and the receiver to capture the link. Whenever an ACK is received by the sender, w is incremented by one and l is incremented by two (one for the packet that has been acknowledged and the other owing to the increase in the congestion window size). Thus, the transition from $[s, w, l]$ to $[s, w+1, l+2]$ takes place with a probability $q\delta U_a$ (δU_a when $l = 0$). Similarly, when a packet is sent by the sender, the congestion window size remains the same but l is decremented by one. Hence, the transmission of a packet by the sender is marked by the transition from $[s, w, l]$ to $[s, w, l-1]$, which occurs with a probability of $p\delta U_t$ (δU_t when $l = w$).

The state transition probabilities of the slow start states are summarized in Table I. The interpretation of the tables in this work is as follows. The first column gives the states involved in the transition, namely the start and end states. It is assumed that both these states are valid. The second column provides the state transition probabilities from the start state to the end state. The third column gives the conditions, if any, for which the probabilities hold. When the state transition probability varies under different conditions, *the first sub-row gives the general conditions that should hold in all cases and the subsequent sub-rows give additional conditions and the corresponding probabilities.*

D. Modeling Congestion Avoidance Phase

This is the predominant phase for long lasting connections. For this phase, we discretise the time into units of RTT. Each round is of a fixed duration, namely RTT. This assumption is also adopted in [1], [12], and [13]. In a round the congestion window size is fixed. Thus the congestion avoidance phase can be considered as consisting of a number of rounds. Let us denote by p_e the probability that a packet is lost, given that either it is the first packet in its round or the preceding

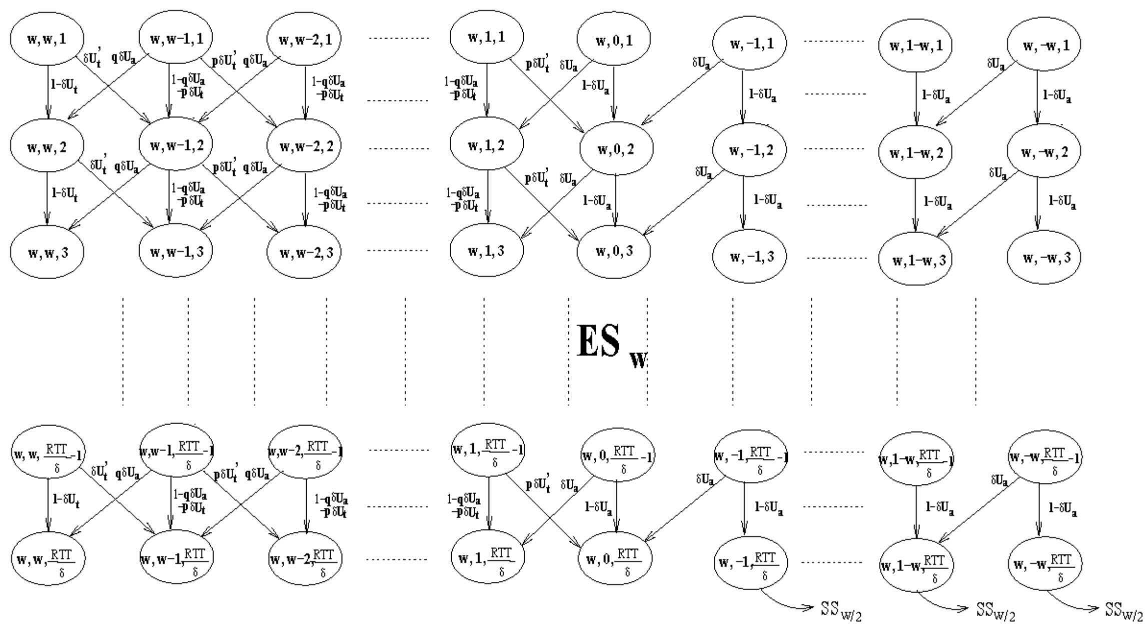

Fig. 2. An Errorless round (ES_w) with congestion window size w. We denote $(1-p_e)U_t$ by $U_t^{'}$ for simplicity of exposition. Here, $SS_{w/2}$ denotes the slow start phase with $SSThresh$ equal to $\frac{w}{2}$.

TABLE I
TRANSITION PROBABILITIES FOR THE SLOW START PHASE

State Transition	*Transition Probability*	*Conditions*
$[s,w,l]$ to $[s,w,l]$	$1-\delta U_t$ $1-\delta U_a$ $1-p\delta U_t - q\delta U_a$	$l=w$ $l=0$ otherwise
$[s,w,l]$ to $[s,w,l-1]$	δU_t $p\delta U_t$	$l=w$ otherwise
$[s,w,l]$ to $[s,w+1,l+2]$	δU_a $q\delta U_a$	$w<s$ $l=0$ otherwise

packet in its round is not lost [1]. We also assume that no ACK/DUPACK can be lost. In the absence of errors, after every round, the congestion window is increased by one. The round in which a loss occurs after a sequence of successful rounds is called the *Penultimate round*. The subsequent round in which either a triple DUPACK occurs or timeout occurs is called the *Ultimate round*. Note that, in our model, we do not assume that the ACKs for the packets sent in a round are received in the same round since it is not necessary that all the packets be sent at the start of the round itself. We model this phase in three steps, namely the *Errorless round*, the *Penultimate round*, and the *Ultimate round*.

Errorless Round

In this type of rounds, after every round the congestion window size is incremented by one. We model every such round by a set of three-dimensional Markov states. The system is said to be in state (w,l,t) if it is in the congestion avoidance phase, where

- w is the current congestion window size, $2 \le w \le W_{Max}$
- l is the number of packets left in the congestion window to be sent, $-w \le l \le w$
- t is the time instance with a granularity of δ, $1 \le t \le \frac{RTT}{\delta}$.

Note that since the minimum value of $SSThresh$ is two, the congestion window size in the congestion avoidance phase can not be less than two. Let us consider the round with congestion window size, w. As shown in Figure 2, every round can be modeled as a $N \times N$ matrix of states, where N is given by $(\frac{RTT}{\delta})(2w+1)$. The t^{th} row of the matrix consists of the states (w,w,t), $(w,w-1,t)$, $(w,w-2,t)$, $\cdots$, $(w,2,t)$, $(w,1,t)$, $(w,0,t)$, $(w,-1,t)$, $(w,-2,t)$, $\cdots$, $(w,-w,t)$. Essentially, it has the states possible in the time interval $[(t-1)\delta, t\delta]$ of a round. The presence of negative values of l is explained while dealing with the Ultimate round. In the absence of errors, within a round the congestion window size remains unchanged and hence, whenever a packet is transmitted by the sender the system goes from state (w,l,t) to $(w,l-1,t+1)$. Similarly, when an ACK is transmitted by the receiver the system goes from state (w,l,t) to $(w,l+1,t+1)$. When the system is in the state $(w,l,\frac{RTT}{\delta})$, in the next δ it increments its congestion window size by one as shown in Figure 3. Note that $1-p_e$ is multiplied to the probability for every successful packet transmission by the sender, since a packet transmitted by the sender is successfully received at the receiver with a probability of $1-p_e$. Table II gives the complete transition probabilities between the states of the Errorless rounds. Note that when the current

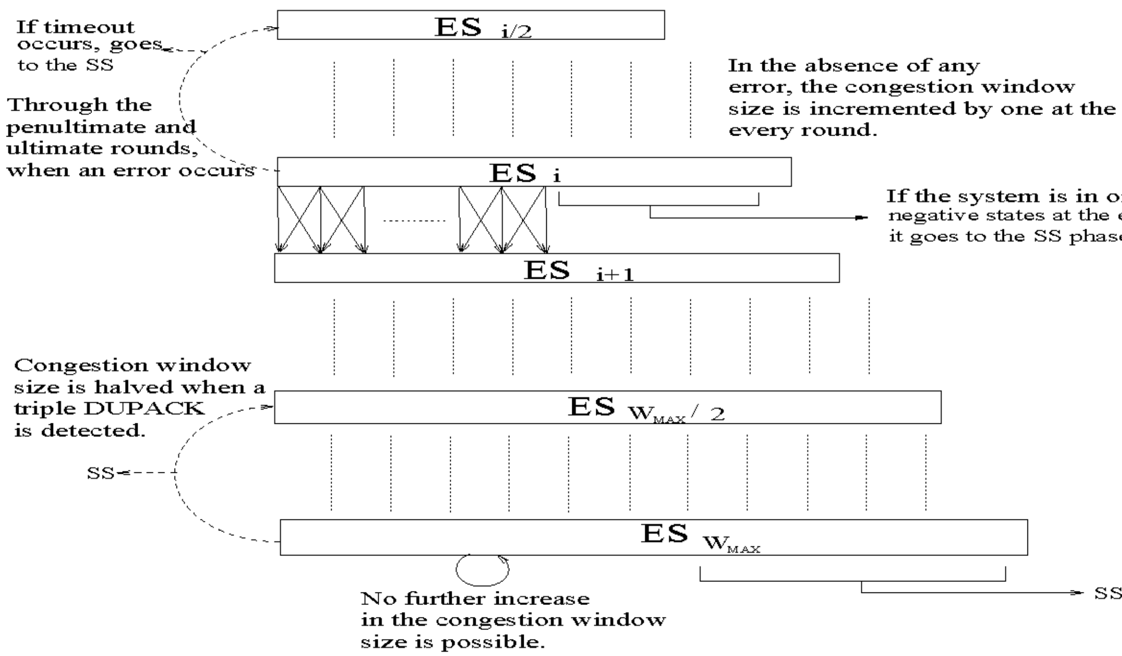

Fig. 3. The transition between the Errorless rounds through the Penultimate and Ultimate rounds for a range of congestion window sizes. The transition between ES_i and ES_{i+1} takes place only at the end of the round.

congestion window size is W_{Max}, no further increase in the congestion window size is possible, so it remains W_{Max} as long as no errors occur, as shown by the self loop in Figure 3.

Penultimate Round

When an error occurs after a sequence of successful (Errorless) rounds, the system enters the Penultimate round. We assume that all the packets in the round after the first loss are also lost. This assumption is justifiable due to the bursty nature of the losses in ad hoc wireless networks. We model the part of this round after the error with the help of a set of four-dimensional Markov states. The system is said to be in state (w, a, l, t) if the system is in the penultimate round, where

- w is the current congestion window size, $2 \leq w \leq W_{Max}$
- a is the number of ACKs awaited, $0 \leq a < w$
- l is the number of packets left in the congestion window to be sent, $0 \leq l < w - a$
- t is the time instance with a granularity of δ, $2 \leq t \leq RTT/\delta$.

Note that the minimum value t takes here is two instead of one, since the system enters into one of these states only after a loss has occurred in one of the Errorless states. Our packet loss model means that, every packet transmitted by the sender when the system is in the state (w, a, l, t) will be lost. As shown in Table III when an error occurs in an Errorless state (w, l, t), the sender will receive $w - l$ ACKs before any DUPACKs is received. So the system goes to the state $(w, w - l, l - 1, t + 1)$. Whenever an ACK is received by the sender the current allowable window, l, is incremented by one and the number of ACKs awaited, a, is reduced by one. Similarly, when a packet is transmitted by the sender, the packet is lost and the current allowable window, l, is decremented by one. However, when both a and l are equal to zero, no event can take place and so the system retains the values of a and l until the end of the round.

Ultimate Round

Since we assume that in the Penultimate round, every packet after the lost packet is also lost, a DUPACK can only be generated by packets sent by the sender in the next round, namely the Ultimate Round. In this round if more than three duplicate acknowledgments are generated, then the congestion window size is halved. However, if a triple DUPACK is not detected, timeout occurs, the $SSThresh$ is set to half of the current congestion window size, and the system goes to the slow start phase with the current window size as one packet. However, since we consider in-order delivery of packets and ACKs, any DUPACK can be received only after all the ACKs, if any, from the previous round are received. Thus, we model the Ultimate round by two sets of five-dimensional states. The system is present in the first set of states, $[w, a, p_w, l, t]$, until all the ACKs from the previous round are received, after which it goes to the second set of states, (w, p_w, l, d, t). The system is said to be in the state $[w, a, p_w, l, t]$ if it is in the ultimate round, where

- w is the current congestion window size, $2 \leq w \leq W_{Max}$
- a is the number of ACKs awaited from the previous round, $1 \leq a < w$
- p_w is the permissible window size (which can increase

TABLE II

TRANSITION PROBABILITIES OF THE ERRORLESS STATES

State Transition	***Transition Probability***	***Conditions***
(w,l,t) to $(w,l,t+1)$	 $1-\delta U_a$ $1-\delta U_t$ $1-p\delta U_t-q\delta U_a$	$t<\frac{RTT}{\delta}$ $l\leq 0$ $l=w$ otherwise
(w,l,t) to $(w,l+1,t+1)$	 δU_a $q\delta U_a$	$t<\frac{RTT}{\delta}$ $l\leq 0$ otherwise
(w,l,t) to $(w,l-1,t+1)$	 $(1-p_e)\delta U_t$ 0 $p(1-p_e)\delta U_t$	$t<\frac{RTT}{\delta}$ $l=w$ $l\leq 0$ otherwise
$(w,l,\frac{RTT}{\delta})$ to $(w+1,l+1,1)$	 $(1-\delta U_a)$ 0 $(1-\delta U_t)$ $(1-p\delta U_t-q\delta U_a)$	$w<W_{Max}$ $l=0$ $l<0$ $l=w$ otherwise
$(w,l,\frac{RTT}{\delta})$ to $(w+1,l+2,1)$	 δU_a 0 $q\delta U_a$	$w<W_{Max}$ $l\in\{0,-1\}$ $l<-1$ otherwise
$(w,l,\frac{RTT}{\delta})$ to $(w+1,l,1)$	 $(1-p_e)\delta U_t$ 0 $p(1-p_e)\delta U_t$	$w<W_{Max}$ $l=w$ $l\leq 0$ otherwise
$(W_{Max},l,\frac{RTT}{\delta})$ to $(W_{Max},l,1)$	$(1-\delta U_a)$ 0 $(1-\delta U_t)$ $(1-p\delta U_t-q\delta U_a)$	$l=0$ $l<0$ $l=w$ otherwise
$(W_{Max},l,\frac{RTT}{\delta})$ to $(W_{Max},l+1,1)$	δU_a 0 $q\delta U_a$	$l\in\{0,-1\}$ $l<-1$ otherwise
$(W_{Max},l,\frac{RTT}{\delta})$ to $(W_{Max},l-1,1)$	$(1-p_e)\delta U_t$ 0 $p(1-p_e)\delta U_t$	$l=w$ $l\leq 0$ otherwise

as packets from the previous round are acknowledged) as above, $0\leq p_w<w-a$

- l is the number of packets that are left in the permissible window to be sent, $0\leq l\leq p_w$
- t is the time in the granularity of δ, $1\leq t\leq\frac{RTT}{\delta}$.

As shown in Table IV, at the end of the Penultimate round if there are still ACKs to be received, in the next δ, the system changes the state from $(w,a,l,\frac{RTT}{\delta})$ to $[w,a,l,l,1]$ if no event takes place or to $[w,a,l-1,l-1,1]$ if a packet is transmitted by the sender or to $[w,a-1,l+1,l+1,1]$ if an ACK is transmitted. Also if an error occurs at the end of an Errorless round, then the system goes to the corresponding state, as shown in Table V, depending on whether ACKs are expected from the previous round. As is clear from Table IV, whenever an ACK is received, a is decremented by one, and p_w and l are each incremented by one. Similarly, whenever a packet is sent by the sender, l is decremented by one. The system remains in these states as long as all the ACKs from the previous round are received. If before the end of this round, all the ACKs have been received, then the system goes to the next set of states, (w,p_w,l,d,t). The system is said to be in state

TABLE III

TRANSITION PROBABILITIES OF THE PENULTIMATE STATES

State Transition	***Transition Probability***	***Conditions***
(w,l,t) to $(w,w-l,l-1,t+1)$	 $p_e\delta U_t$ $p_ep\delta U_t$	$t<\frac{RTT}{\delta}$ $l=w$ otherwise
$(w,0,0,t)$ to $(w,0,0,t+1)$	1	$t<\frac{RTT}{\delta}$
(w,a,l,t) to $(w,a-1,l+1,t+1)$	 δU_a $q\delta U_a$	$w\neq a$ & $t<\frac{RTT}{\delta}$ $l=0$ otherwise
(w,a,l,t) to $(w,a,l-1,t+1)$	 δU_t $p\delta U_t$	$t<\frac{RTT}{\delta}$ $a=0$ otherwise
(w,a,l,t) to $(w,a,l,t+1)$	 $1-\delta U_t$ $1-\delta U_a$ $1-p\delta U_t-q\delta U_a$	$t<\frac{RTT}{\delta}$ $a=0$ $l=0$ otherwise

(w,p_w,l,d,t), if the system is in the Ultimate round and all the ACKs from the previous round have been received, where

- w denotes the congestion window size at which this state is entered, $2\leq w\leq W_{Max}$
- p_w denotes the permissible window size which is less than w owing to the fact that some packets were lost in the previous round, $0\leq p_w<w$
- l denotes the number of packets that are left in the permissible window to be sent, $0\leq l\leq p_w$
- d denotes the number of duplicate acknowledgments that were received, $0\leq d\leq 3$
- t denotes the time in the granularity of δ, $1\leq t\leq\frac{RTT}{\delta}$.

If at the end of the Penultimate round, the system is in states of the form, $(w,0,l,\frac{RTT}{\delta})$, no more ACK are awaited. Hence, the system goes directly to either $(w,l-1,l-1,0,l)$ or $(w,l,l,0,1)$ depending on whether a packet is transmitted or not. Similarly if the system is in states of the form, $(w,1,l,\frac{RTT}{\delta})$, where only one ACK is awaited, then if in the next δ, the sender receives an ACK then the system goes directly to the state, $(w,l+1,l+1,0,1)$. When the system is in one of the states of the form, (w,p_w,l,d,t), if the sender receives a DUPACK both l and d are incremented by one. However, if the sender transmits a packet, the number of packets that can be sent from the permissible window, l, is decremented by one, as shown in Table IV. The system waits until the triple DUPACK detection takes place or until the end of the round whichever is earlier. When it occurs the congestion window is halved and the transition as shown in Table VI takes place. Since the congestion window is made half, there exists a possibility that more packets than that permitted by the updated congestion window size would have been sent in the Ultimate round. This explains the negative states of the form $(w,-1,t),(w,-2,t),\cdots,(w,-w,t)$. As shown in Table II, when the system is in one of the negative states, the only event that can occur is the transmission of an ACK by the receiver. Thus, the only transitions possible are from (w,l,t) to $(w,l+1,t)$ (when an ACK is received at the sender) with a probability of δU_a and (w,l,t) to $(w,l,t+1)$

TABLE IV

TRANSITION PROBABILITIES FOR THE ULTIMATE ROUND WHEN THE CONGESTION WINDOW SIZE IS w

State Transition	***Transition Probability***	***Conditions***
$[w, a, p_w, l, t]$ to $[w, a, p_w, l, t+1]$	 $1 - \delta U_t$ $1 - \delta U_a$ $1 - p\delta U_t - q\delta U_a$	$t < \frac{RTT}{\delta}$ $l = p_w$ $l = 0$ otherwise
$[w, a, p_w, l, t]$ to $[w, a, p_w, l-1, t+1]$	 δU_t $p\delta U_t$	$t < \frac{RTT}{\delta}$ $l = p_w$ otherwise
$[w, a, p_w, l, t]$ to $[w, a-1, p_w+1, l+1, t+1]$	 δU_a $q\delta U_a$	$t < \frac{RTT}{\delta}$ and $a \geq 2$ $p_w = 0$ otherwise
$[w, 1, p_w, l, t]$ to $(w, p_w+1, l+1, 0, t+1)$	 δU_a $q\delta U_a$	$t < \frac{RTT}{\delta}$ $p_w = 0$ otherwise
$(w, 0, 0, 0, t)$ to $(w, 0, 0, 0, t+1)$	1	$t < \frac{RTT}{\delta}$
(w, p_w, l, d, t) to $(w, p_w, l, d, t+1)$	 $1 - \delta U_t$ $1 - \delta U_a$ $1 - p\delta U_t - q\delta U_a$	$t < \frac{RTT}{\delta}$ $l = p_w$ $l = 0$ otherwise
(w, p_w, l, d, t) to $(w, p_w, l-1, d, t+1)$	 δU_t $p\delta U_t$	$t < \frac{RTT}{\delta}$ $l = p_w$ otherwise
(w, p_w, l, d, t) to $(w, p_w, l+1, d+1, t+1)$	 δU_a $q\delta U_a$	$t < \frac{RTT}{\delta}$ and $d \leq 2$ $l = 0$ otherwise

(when no event occurs) with a probability of $1 - \delta U_a$.

E. Transition between the Slow Start Phase and Congestion Avoidance Phase

While the system is in the slow start phase, if the current congestion window excess the $SSThresh$, as shown in Table VI, the system goes from the slow start state, $[s, s, l]$, to one of the Errorless round state, $(s+1, l+2, 1)$. That is the system goes into one of the Errorless states. If by the end of the Ultimate round a triple DUPACK indication does not occur then, as shown in Table VI, timeout occurs and the system makes its $SSThresh$ equal to half of the current congestion window, sets the congestion window size as one packet and goes to the slow start phase. Also if at the end of the round, the system remains in one of the negative states corresponding to the Errorless round, we assume that timeout occurs and the system goes to the slow start phase, as shown in Figure 3.

To sum it up, Figure 4 gives the transitions amongst the slow start phase and the different rounds of congestion avoidance phase.

F. Assumptions (Re-Visited)

We summarized the various assumptions in the above modeling of TCP over ad hoc networks. The justifications for the various assumptions have already been stated earlier.

- Every node in the network under consideration is assumed to have the same transmission range, carrier sense range, and interference range.
- No out-of-order packet/ACK delivery can occur.
- We assume that ACKs/DUPACKs cannot be lost.

TABLE V

TRANSITION PROBABILITIES BETWEEN THE PENULTIMATE ROUND AND ULTIMATE ROUND

State Transition	***Transition Probability***	***Conditions***
$(w, a, l, \frac{RTT}{\delta})$ to $[w, a, l, l, 1]$	 $(1 - \delta U_a)$ $(1 - p\delta U_t - q\delta U_a)$	$a \neq 0$ $l = 0$ otherwise
$(w, a, l, \frac{RTT}{\delta})$ to $[w, a, l-1, l-1, 1]$	$p\delta U_t$	$a \neq 0$
$(w, a, l, \frac{RTT}{\delta})$ to $[w, a-1, l+1, l+1, 1]$	 δU_a $q\delta U_a$	$a > 1$ $l = 0$ otherwise
$(w, 0, 0, \frac{RTT}{\delta})$ to $(w, 0, 0, 0, 1)$	1	
$(w, 0, l, \frac{RTT}{\delta})$ to $(w, l, l, 0, 1)$	$1 - \delta U_t$	
$(w, 0, l, \frac{RTT}{\delta})$ to $(w, l-1, l-1, 0, 1)$	δU_t	
$(w, 1, l, \frac{RTT}{\delta})$ to $(w, l+1, l+1, 0, 1)$	 δU_a $q\delta U_a$	 $l = 0$ otherwise
$(w, l, \frac{RTT}{\delta})$ to $[w, w-l, l-1, l-1, 1]$	$p_e p\delta U_t$	$l < w$
$(w, w, \frac{RTT}{\delta})$ to $(w, w-1, w-1, 0, 1)$	$p_e\delta U_t$	
$(w, p_w, l, 3, t)$ to $(\lfloor w/2 \rfloor, \lfloor w/2 \rfloor - p_w + l, 1)$	 δU_a $q\delta U_a$	$t < \frac{RTT}{\delta}$ $l = 0$ otherwise

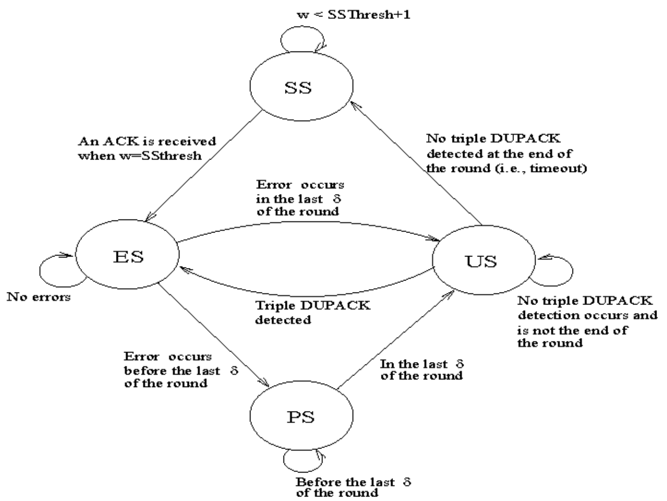

Fig. 4. State transition between the different rounds. Here SS, ES, PS, and US denotes the Slow start, Errorless, Penultimate, and Ultimate states, respectively.

TABLE VI
TRANSITION PROBABILITIES BETWEEN THE SLOW START PHASE AND THE CONGESTION AVOIDANCE PHASE

State Transition	***Transition Probability***	***Conditions***
$(2w, p_w, l, d, \frac{RTT}{\delta})$ to $[w, 1, 1]$	1	
$(2w+1, p_w, l, d, \frac{RTT}{\delta})$ to $[w, 1, 1]$	1	
$(2w, l, \frac{RTT}{\delta})$ to $[w, 1, 1]$	1	$l < 0$
$(2w+1, l, \frac{RTT}{\delta})$ to $[w, 1, 1]$	1	$l < 0$
$[s, s, l]$ to $(s+1, l+2, 1)$	δU_a $q\delta U_a$	$l = 0$ otherwise
$[2w, p_w, l, d, \frac{RTT}{\delta}]$ to $[w, 1, 1]$	1	

- In the Penultimate round we assume that all the packets in the round after the first loss are also lost.
- Finally, in the slow start phase and the DUPACK generating round, i.e. the Ultimate round, we assume that no packets are lost.

III. GOODPUT CALCULATION

Goodput is defined as the number of packets that successfully reach the receiver in unit time. To determine the goodput from the above model, we have to find the steady state probabilities of the above states. First, we proceed by proving that stationary states do exist in our case and then we give the goodput equation.

Lemma 3.1: The Markov chain thus formed is aperiodic and irreducible and hence has stationary solutions.

Proof: It is sufficient to prove that there exists a n^* for which T^{n^*} has all non-zero entries, where T is the transition matrix [10]. First, note that for every pair of states e_1 and e_2 there exists an m such that in m steps, the system can go from e_1 to e_2, provided p_e is not zero or one. Now consider a state $e_3 \in \{[s, w, l] \mid s,\ w,\ and\ l\ are\ valid\}$. Since for every e_1 and e_2, there exists an m_1 and m_2 such that in m_1 (m_2) steps we can go from e_1 (e_3) to e_3 (e_2), there exists a path of length $n = m_1 + m_2$ between e_1 and e_2 that goes through e_3. Consider the maximum such n over all pairs e_1 and e_2, n^*. Now we can conclude that T^{n^*} has all positive entries, since for every pair e_1 and e_2 whose corresponding n is less than n^*, make the system stay in e_3 for $n^* - n$ steps using the self loop on e_3. Thus, T^{n^*} has all non-zero entries. ■

A. Goodput Equation

Let us denote the steady state probability of the states $[s, w, l]$, (w, l, t), $[w, a, p_w, l, t]$, and (w, p_w, l, d, t) by $\Omega(s, w, l)$, $\Pi(w, l, t)$, $\Gamma(w, a, p_w, l, t)$, and $\Theta(w, p_w, l, d, t)$ respectively. From these variables, the goodput (GP) can be calculated from the following equation.

$$GP = \sum_{l>0} p_1(w, l)\Omega(s, w, l) + \sum_{l>0} p_2(w, l)\Pi(w, l, t) + \sum_{l>0} p_3(p_w, l)\Gamma(w, a, p_w, l, t) + \sum_{l>0} p_3(p_w, l)\Theta(w, p_w, l, d, t) \quad (1)$$

where

$$p_1(w, l) = \begin{cases} U_t & \text{if } l = w \\ pU_t & \text{otherwise} \end{cases}$$

$$p_2(w, l) = \begin{cases} U_t & \text{if } l = w \\ p(1 - p_e)U_t & \text{otherwise} \end{cases}$$

$$p_3(p_w, l) = \begin{cases} U_t & \text{if } l = p_w \\ pU_t & \text{otherwise} \end{cases}$$

Note that, in the goodput calculation, we have excluded the Penultimate round states and considered only the Slow Start states, the Errorless round states, and the Ultimate round states. This is because we have assumed that all the packets that are transmitted by the sender are lost when the system is in one of the four-dimensional states, (w, a, l, t), that represent the Penultimate round. The throughput calculation can be done on similar lines by including the Penultimate round in the calculation.

IV. DETERMINING THE PARAMETERS

The values that the parameters U_t, U_a, p, and q take, depend on the number of hops and the link bandwidth. Channel spatial reuse and local channel contention affect the values that these parameters take. As described earlier we need to calculate only the value of U_t, since the value of U_a can be easily calculated from the expression $\frac{U_t}{U_a} = \frac{size\ of\ ACK}{size\ of\ packet}$. The value of U_t is given by

$$U_t = \frac{j(M+1)^2 + (r-j)M^2}{(rM+j)^2} \times \frac{1}{T} \quad (2)$$

where $N = rM + j$, $M \geq 0$ and $j \in \{1, 2, 3, \cdots, r\}$ and $p = q$ is given by

$$q = \begin{cases} \frac{1}{2h} & \text{if } h < r+1 \\ \frac{1}{2r} & \text{otherwise} \end{cases}$$

where h is the number of hops and r is given by $\lceil InRange/d \rceil + 1$. $InRange$ denotes the interference range of each node in the network under consideration. The derivation of the above expressions is similar to that given in [2] and has been omitted due to space constraint.

V. Comparison of the Model and the simulation results

We use ns-2 to simulate TCP over a N-hop chain network. Note that by considering a chain topology, amongst other things, we exclude the possibility of out-of-order packer delivery. FTP is used as the application layer protocol to generate TCP traffic. The default configuration values were considered. The number of nodes in the chain was varied. The adjacent nodes in the chain are $200m$ apart. The channel capacity is assumed to be $2Mbps$ and the nominal transmission radius is $250m$. Nodes have a carrier sense radius of $550m$ and interference range ($InRange$) of $550m$. Note that in this setting the value of r is four. The MAC and routing protocol used in our simulations are IEEE 802.11 and AODV, respectively. The value of the size of a packet and ACK are taken as 1460 $bytes$ and 40 $bytes$, respectively. Each simulation is run for 900 *seconds* and the goodput is calculated between 50 *seconds* to 900 *seconds*. Each point in the graphs is obtained out of results from thirty runs, each with a different seed. All the results in this section have been obtained at 95% confidence level. The theoretical results are obtained using *matlab*. In order to get the results in *matlab*, we need the value of $\frac{1}{T}$, where T was earlier defined as the average transmission time for one packet over one hop link. However, its value depends on the channel contention, MAC and routing overheads, and link bandwidth. Hence, we decide the value of $\frac{1}{T}$ based on simulation results. We have also considered various values of W_{Max}. However, the results have no significant variations for various values of W_{Max}.

We use two error models to evaluate our analytical model[1]. The first error model is a uniform error model with a probability of loss given by p. We introduce this error at every node of the chain. So every node drops a packet with a probability p. Thus the effective end-to-end loss probability, p_e, is given by $1-(1-p)^N$, where N is the number of hops. By introducing the error model at every node, we are effectively covering a large range of end-to-end loss probabilities. The second error model is a two state Markov error model, where the probability of going from the good to bad state is p and the probability of going from the bad to good state is q. The motive behind the choice of this error model is ingrained in the type of errors we have considered in the theoretical model. The two state Markov error model captures the assumption that every packet in the round after the first lost packet is also lost. We introduce this error model at the receiver to see the effect of q as a function of N.

Figure 5 shows the comparison of simulation and theoretical results of the TCP goodput as the number of hops changes from 1 to 12, when the uniform error model is introduced at every node. Here, we consider the value of p to be 0.01. The theoretical results are presented for two different values of $\frac{1}{T}$, $1200kbps$ and $600kbps$, where the end-to-end error probability, p_e, is taken as $1-(1-p)^N$. Note that when the number of hops is less than four, the simulation results closely match with the theoretical results corresponding to $\frac{1}{T} = 1200kbps$, and as the number of hops increase they are closer to the theoretical results corresponding to $\frac{1}{T} = 600kbps$. This can be explained by the fact that in our simulation setting $r = 4$ and for $N \leq r$ no link is simultaneously used due to the hidden terminal problem. However, when $N > r$, although channel spatial reuse improves the utilization, the higher local contentions will have an opposite effect. Note that as the number of hops increases, the effect of the MAC and routing protocols on the goodput increases, as observed in [7]. This effect is further increased in the presence of errors, as is clear from the gap between the theoretical (for $\frac{1}{T} = 600kbps$) and simulation results as the number of hops increases. This is because, in the presence of errors, a large amount of MAC and routing overhead (which include re-transmission at the MAC layer and false route error messages) is introduced in the system. This increases the time, T, it takes to service a packet which in turn decreases the effective bandwidth, $\frac{1}{T}$, available.

Figure 6 considers the case when $p = 0.05$. Note that although similar behavior is observed in this case, we notice the gap between the theoretical results with $\frac{1}{T} = 600kbps$ and the simulation results at higher hop lengths, is more in this case. This is due to the fact that as the number of hops increases, the effective end-to-end error probability increases drastically. For example, in our case when $p = 0.05$ the effective end-to-end error probability is $\simeq 0.05$, $\simeq 0.14$, $\simeq 0.26$, $\simeq 0.37$, and $\simeq 0.46$ when the number of hops is 1, 3, 6, 9, and 12, respectively. Hence, nearly half of the packets transmitted are lost when the number of hops is 12. This high rate of packet loss further increases the MAC and routing overhead.

Figures 7 and 8 show the comparison of simulation and theoretical results of the TCP goodput as the number of hops increase, when a two state Markov error model is introduced at the receiver, with the value of p as 0.01 and 0.05, respectively. We consider the values of q from the set $\{1, 0.95, 0.9, 0.85, 0.8, 0.75\}$. Again, we plot the theoretical results corresponding to the values of $\frac{1}{T}$, $1200kbps$ and $600kbps$, where p_e is taken to be p. We observe the similar behavior even in this case. However, one interesting observation is that the effect of the values of q considered

[1]The p and q used here to describe the error models, are different from those used in the previous sections.

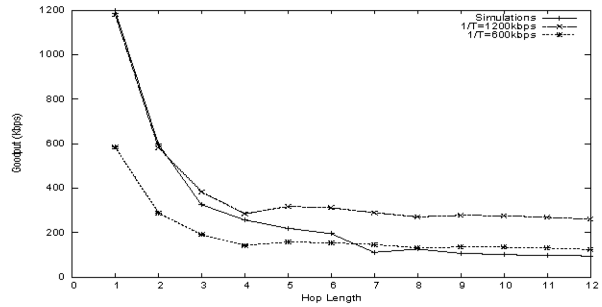

Fig. 5. Goodput vs Hop Length (for fixed p=0.01) with uniform error model.

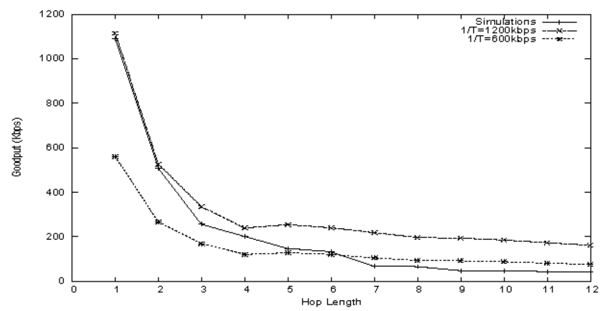

Fig. 6. Goodput vs Hop Length (for fixed p=0.05) with uniform error model.

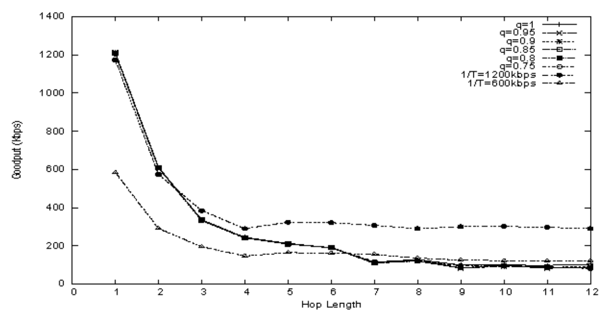

Fig. 7. Goodput vs Hop Length (for fixed p=0.01) with two state Markov error model.

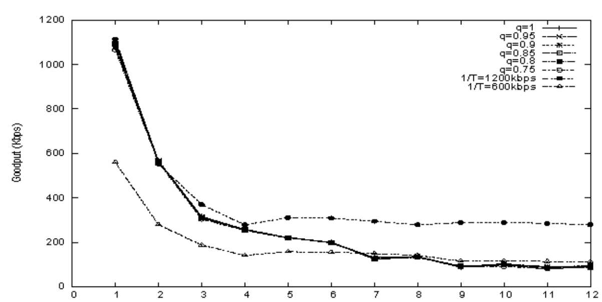

Fig. 8. Goodput vs Hop Length (for fixed p=0.05) with two state Markov error model.

here, on the goodput decreases as the hop length increases. One common observation, which is in agreement with [2], is that the goodput decreases very fast initially as a function of the number of hops and stabilizes at a larger number of hops.

VI. Conclusions and Future Work

In this work, we proposed a multi-dimensional Markovian model for the performance of TCP over ad hoc wireless networks. This model captures all the essential features of TCP and the effect of IEEE 802.11. We considered both the slow start phase and the congestion avoidance phase in our model formulation. Multi-dimensional Markov chains are used to model each of these phases along with the transitions between them. We used the resulted steady state probabilities to calculate the goodput. Further, we corroborated the theoretical results with simulation results. We observed that the effect of the MAC layer on the performance of TCP increases with the number of hops and with increased errors.

A number of directions for future work remain to be explored. One such direction is to include the behavior of MAC and the routing protocols, which will help in providing a more accurate model and in determining the value of $\frac{1}{T}$ theoretically instead of depending on simulations. Another direction is to extend this model to hold for multiple TCP flows. Also, in our model we have assumed that all the packets in the round that follow a lost packet are also lost. It would be of interest to remove this assumption and bring in various other error models while modeling the system.

References

[1] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Throughput: A Simple Model and its Empirical Validation", *in Proc. ACM SIGCOMM 1998*, pp. 303-314, September 1998.

[2] H. Xiao, K. C. Chua, J. A. Malcolm, and Y. Zhang, "Theoretical Analysis of TCP Throughput in Ad hoc Wireless Networks", *in Proc. IEEE GLOBECOM 2005*, vol. 5, pp. 2714-2719, November 2005.

[3] L. L. Peterson and B. S. Davie, *Computer Networks*, 3rd Edition, Morgan Kaufmann Publishers, 2003.

[4] C. Siva Ram Murthy and B. S. Manoj, *Ad Hoc Wireless Networks: Architectures and Protocols*, Prentice Hall, New Jersey, 2004.

[5] M. Gerla, R. Bagrodia, L. Zhang, K. Tang, and L. Wang, "TCP over Wireless Multi-hop Protocols: Simulations and Experiments", *in Proc. IEEE ICC 1999*, pp. 1089-1094, June 1999.

[6] G. Holland and N. H. Vaidya, "Analysis of TCP Performance over Mobile Ad hoc Networks", *in Proc. IEEE/ACM MOBICOM 1999*, pp. 219-230, August 1999.

[7] S. Bansal, R. Shorey, and A. A. Kherani, "Performance of TCP and UDP Protocols in Multi-hop Multi-rate Wireless Networks", *in Proc. WCNC 2004*, vol. 1, pp. 231-236, March 2004.

[8] M. Prabakaran, A. Mahasenan, and A. Kumar, "Analysis and Enhancement of TCP Performance over an IEEE 802.11 Multi-hop Wireless Network: Single Session Case", *in Proc. ICPWC 2005*, pp. 29-33, January 2005.

[9] A. A. Kherani and R. Shorey, "Throughput Analysis of TCP in Multi-hop Wireless Networks with IEEE 802.11 MAC", *in Proc. WCNC 2004*, vol. 1, pp. 237-242, March 2004.

[10] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, 4th Edition, McGraw-Hill, 2002.

[11] D. P. Bertsekas and R. Gallager, *Date Networks*, 2nd Edition, Prentice Hall, New Jersey, 1992.

[12] J. Mahdavi and S. Floyd, "TCP-friendly Unicast Rate-based Flow Control", *Note in end2end-interest mailing list*, January 1997.

[13] M. Mathis, J. Semske, J. Mahdavi, and T. Ott, "The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm", *ACM Computer Communication Review*, vol. 27, no. 3, pp. 67-82, July 1997.