This full text paper was peer reviewed at the direction of IE... ...2005 proceedings.

Theoretical A... ...Adhoc

...en the source and destination

...nized as follows. The system ...formance in adhoc wireless ...n II, and the Markov chain ...of the system is discussed in section III. Section ...ts the analysis of the TCP throughput over the ...ology from specific cases to the general case. Section ...nstrates comparison of the simulation and numerial ... Finally section VI concludes the paper.

II. System Model

...nsider the simple adhoc wireless network scenario shown ...ig. 1. $N+1$ nodes form a *string* with length N. Each node ... the same transmission radius, the same carrier sense radius, ...d the same interference radius. One TCP connection is run ...om node 0 to node N crossing all the intermediate nodes in ...he *string*.

We model the communication process of the TCP connection as in Fig. 2 after analyzing the trace files from simulations carefully. The source and destination nodes both have a FIFO

... of ... in ...ring ...ases ... then ...a gave ... would ...t giving ... analysis ...logy was ...ling in [8]

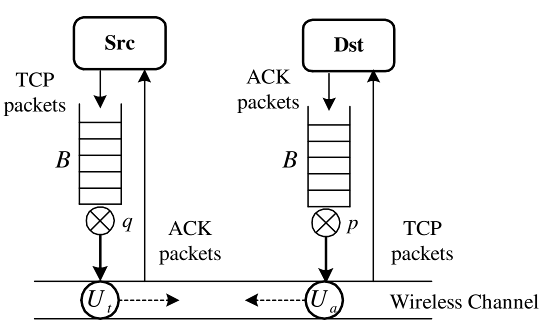

Fig. 2. System model of TCP in wireless adhoc networks

forward buffer of size B. The source has infinite data to send, so that TCP packets are always of the maximum packet size. For each packet that is received by the destination, a cumulative acknowledgement (ACK) is generated which can be modelled as containing the next expected segment number. TCP packet service rate is U_t, the typical number of TCP packets the system serves per unit time when it is constantly busy. ACK packet service rate is U_a, the typical number of ACK packets the system serves per unit time when it is constantly busy. The "unit time" includes TCP/ACK packet transmission delay, propagation delays, processing delays at the nodes from layer to layer and the average MAC layer contention delays. The TCP/ACK packet service rates are affected by the spatial reuse of wireless channel in the *string* topology. As all nodes compete to use the channel, the chance for any node to send packet is determined by the underlying MAC protocol and traffic distribution. Generally, we denote the *average* probability of the source to send a TCP packet as q, and the *average* probability of the destination to send an ACK packet as p. The reason to use the average probability of the node accessing the channel is because the performance metric of interest is the average TCP throughput.

The system model captures the unique communication features of TCP in adhoc wireless networks including link layer channel contention and channel spatial reuse. It is similar to but different from system models used in analyzing TCP performance in other kind of networks [3] [7].

III. Markov Chain Modelling

We use Markov chain modelling to analyze the proposed system model. Firstly, a discrete Markov chain is formed for the case when the source and the destination are single-hop away. The Markov chain is then extended to the general case when the source and the destination are multiple hops away. The assumptions made are as follows:

- There is no random loss of packets due to channel error. The channel appears error-free to the upper layer because of the error coding schemes and link layer retransmission protocols.
- The channel also appears collision-free to the upper layer because the MAC layer protocol is collision avoided, such as MACAW [2].
- The maximum TCP congestion window size is less than the buffer size at each node.
- There is no packet dropping due to buffer overflow, and no packet loss due to link layer contention. This assumption is supported by simulation results demonstrating that packet loss due to buffer overflow is rare and packet loss due to link contention is very low for single TCP over a *string* topology.
- The distribution of inter-service time of TCP packets and ACK packets are exponentially distributed which include packet retransmissions. This assumption is based on the fact that exponential backoff during contention is commonly used in MAC protocols, e.g., IEEE 802.11.

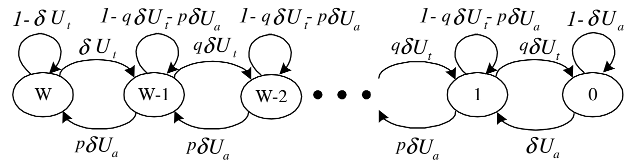

Fig. 3. Transition diagram when the number of hop is 1 and TCP maximum congestion window size is W.

A. Single-hop Case

There are only TCP packets at the source and only ACK packets at the destination. Either the source or the destination sending packets will change the number of the packets queued in their buffers. Ignoring the slow start phase which is a small part of the data transmission, the sum of the TCP packets at the source and the ACK packets at the destination equals to the TCP congestion window size in packets. As it is assumed that no packet loss exists due to channel error, buffer overflow, or link contention, the TCP connection thus increases its congestion window size to the maximum value and stabilizes there.

Discrete-time Markov chain is applied to analyze TCP performance here. Let us focus attention at times 0, δ, 2δ, $\ldots$, $k\delta$, $\ldots$ as in [1] (pages 162-173), where δ is a small positive number. Let F denote as the number of TCP packets in the source node buffer at time $k\delta$. F thus is a Markov chain on the state-space $\{m : W \geq m \geq 0\}$, where W is the maximum TCP congestion window size. The transition diagram of the Markov chain is shown in Fig. 3.

When there are W TCP packets in the source node, i.e., the first state in Fig. 3, the destination node does not have ACK packets to send. Therefore, the source node catches the wireless channel for sure. The resulting transition probability from the state of $F = W$ to $F = W - 1$ is δU_t, where U_t is the service rate of TCP packets and the inter-service time is assumed to be exponentially distributed. Similarly, the transition probability from the state of $F = 0$ to $F = 1$ is δU_a, where U_a is the service rate of ACK packets and the inter-service time is assumed to be exponentially distributed.

When there are F $(W - 1 \geq F \geq 1)$ TCP packets in the source, there are $W - F$ ACK packets in the destination node. The source node and destination node compete with each other to use the channel. The source node has average probability of q to access the channel (see Fig. 2), the transition probability from the state of $F = W - i$ to $F = W - i - 1$ is thus $q\delta U_t$, where $1 \leq i \leq W - 1$. Similarly, the transition probability from the state of $F = W - i$ to $F = W - i + 1$ is $p\delta U_a$, where p is the average probability of the destination to access the channel.

B. Multiple-hop Case

When the source and destination are multiple hops away, an accurate Markov model at times $k\delta$ would be on the state space $\{(F_0, A_0), (F_1, A_1), \ldots, (F_i, A_i), \ldots, (F_N, A_N)\}$, where (F_i, A_i) are the numbers of TCP and ACK packets at node i. This multi-dimensional Markov chain modelling

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2005 proceedings.

$$\mathbf{Q} = \begin{pmatrix} 1-\delta U_t & \delta U_t & 0 & 0 & \dots & 0 \\ p\delta U_a & 1-p\delta U_a - q\delta U_t & q\delta U_t & 0 & \dots & 0 \\ 0 & p\delta U_a & 1-p\delta U_a - q\delta U_t & q\delta U_t & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & p\delta U_a & 1-p\delta U_a - q\delta U_t & q\delta U_t \\ 0 & \dots & 0 & 0 & \delta U_a & 1-\delta U_a \end{pmatrix} \quad (1)$$

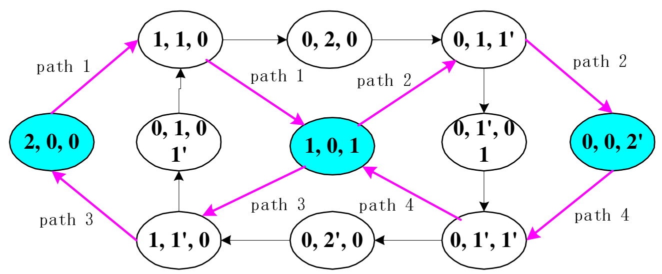

Fig. 4. Transition diagram when the number of hop is 2 and the maximum TCP congestion window size is 2 packets. The number of TCP packets at nodes are shown as 1 or 2; the number of ACK packets are shown as 1' or 2'. (1, 1', 0) denotes that there is 1 TCP packet at the source node, 1 ACK packet at the middle node, and none packet at the destination node.

considers the number of TCP packets and ACK packets in the source, the destination and the intermediate nodes along the *string* topology. Unfortunately, such a modelling is difficult to tackle. For example, even when the *string* topology is 2-hops long, and the TCP maximum congestion window size is 2 packets, the Markov chain already has 11 states as shown in Fig. 4, and the number of states increases quickly with the increase of the TCP maximum congestion window size and the length of the *string*. This happens due to the sharing of wireless channel between nodes. When two nodes within interference range both have packets to send, there are two possible next states. The more states the Markov chain has, the more difficult it is to solve. A heuristic alternative is applied as follows.

The more states the Markov chain has, the more paths are available in the transition diagram for a TCP packet to be transmitted from the source to the destination, or for an ACK packet from the destination to the source. Whatever states the packet goes through in between, it finally passes all the nodes of the *string*. Let us denote the path for a TCP packet to be transmitted from the source to the destination as a *forward* path; and the path for an ACK packet to be transmitted from the destination to the source as a *backward* path. A transmission round of TCP is composed of one *forward* path and one *backward* path.

The average utilization of the *string* topology should be the average of all the transmission rounds. Intuitively, ***for each forward path, a corresponding complementary backward path exists which makes this round of transmission to be exactly the average value.*** With this understanding, the analysis can be sought out based on a specific transmission round with a pair of complementary *forward* and *backward* paths only. All the TCP packets go through the specific *forward* path, and all the ACK packets go through the specific *backward* path in such analysis.

A simple transmission round is chosen as follows: for the *forward* path, when the source node gets chance to start sending a TCP packet, the packet passes all the intermediate nodes continuously until it reaches the destination; the *forward* path can be path 1 or path 2 in Fig. 4. Likewise, for the *backward* path, when the destination node gets chance to start sending an ACK packet, the ACK packet passes all the intermediate nodes continuously until it reaches the source; the *backward* path can be path 3 or path 4 in Fig. 4. These paths are *complementary*.

In the above specific transmission round, once one packet in the source node is sent out, the number of packets in the buffer of source node decreases by 1. After the packet goes continuously to the destination, the destination absorbs the TCP packet and generates an ACK packet. The reverse process is the same. Consequently, the changing of number of packets in the source and the destination is the same as in the single-hop scenario. The Markov modelling of this complete transmission round is thus exactly the same as in Fig. 3 when we only consider the marked paths and states in Fig. 4.

In the following section, we are going to analyze the discrete Markov chain in Fig. 3 for both the single-hop and multiple-hop cases.

IV. Analysis

A. TCP Throughput

The transition matrix $\mathbf{Q}$ of the Markov chain in Fig. 3 is listed in (1). Letting $\pi = (\pi_W, \pi_{W-1}, ..., \pi_0)$ denote the stationary probability distribution of the Markov chain, we have $\sum_{i=0}^{W} \pi_i = 1$, and $\pi = \pi \times \mathbf{Q}$ at steady state. From these it is derived that

$$\pi_W = \frac{1}{1 + \sum_{i=1}^{W-1} \frac{1}{q}(\frac{q}{p}\rho)^i + \frac{p}{q}(\frac{q}{p}\rho)^W} \quad (2)$$

$$\pi_i = \frac{1}{q}(\frac{q}{p}\rho)^{W-i}\pi_W, \qquad 1 \le i \le W-1 \quad (3)$$

where ρ is the ratio of U_t to U_a, i.e., $\rho = \frac{U_t}{U_a}$.

From the transition diagram, we denote the TCP throughput of state i by λ_i, and the average throughput of TCP packets by λ. Thus $\lambda = \sum_{i=0}^{W} \pi_i \lambda_i$, where $\lambda_W = U_t, \lambda_i = qU_t$ when $W-1 \ge i \ge 1$, and $\lambda_0 = 0$. Therefore

$$\lambda = U_t \pi_W \frac{1-(\frac{q}{p}\rho)^W}{1-\frac{q}{p}\rho} \quad (4)$$

The source and destination nodes are located at the two ends of the *string* topology with $N-1$ intervening nodes. The positions of the source and destination are the same, and they also have the same number of packets to send since we assume that one ACK is generated for each TCP packet. Thus, the source and destination nodes have the same *average* probability to access the channel successfully, i.e., $p = q$. (2) and (4) can then be simplified as

$$\pi_W = \frac{1-\rho}{1-\rho^{W+1}+(\frac{1}{q}-1)\rho(1-\rho^{W-1})} \quad (5)$$

$$\lambda = U_t\frac{1-\rho^W}{(1-\rho^{W+1})+(\frac{1}{q}-1)\rho(1-\rho^{W-1})} \quad (6)$$

ρ is the ratio of TCP packet service rate to the ACK packet service rate. Since TCP packets and ACK packets are transmitted at the same channel, and they travel the same number of hops between the source and the destination, ρ is always approximately equal to the ratio of the TCP packet service rate to the ACK packet service rate in the case when there is only one hop between the source and the destination. This is further approximately equal to the ratio of the ACK packet size to the TCP packet size. The TCP packet (say, 1460 bytes) is normally much larger than the ACK packet (say 40 bytes), ρ is thus much smaller than 1 (say, around 40/1460=2.7%), i.e., $\rho \ll 1$. The maximum TCP congestion window size W $pkts$ is an integer and when it is big enough, it is expected that $1-\rho^W \approx 1-\rho^{W-1} \approx 1-\rho^{W+1}$. Thus from (6), the *average* TCP throughput is derived as

$$\lambda = U_t\frac{1}{1+(\frac{1}{q}-1)\rho} \quad (7)$$

In the above analysis, it is shown clearly that the ratio of ACK packet size to TCP packet size being much less than 1 is required to carry out the approximation.

B. TCP Service Rate

We have defined U_t as the service rate of TCP packets seen from the source node and q as the average probability for the source node to access the wireless channel. Their values, however, change with the length of the *string*, i.e., the number of hops that the TCP session crosses from the source to the destination (see Fig. 1). This is because of the effect of global channel spatial reuse and local channel contention in adhoc wireless networks. By definition, service rate is the "typical number of customers the system serves per unit time when it is constantly busy" [1] (page 152). When the number of hops is N, let the average number of TCP packets being transmitted (served) in the system be $\overline{I_N}$ and the service time to transmit a packet be T_N, the definition of service rate gives:

$$U_{tN} = \frac{\overline{I_N}}{T_N} \quad (8)$$

Let q_N be the average probability that the source node accesses the wireless channel when the number of hops is N. As N changes, $\overline{I_N}$, T_N and q_N also change. In the following, we evaluate the values of $\overline{I_N}$, T_N, and q_N as N changes.

1) N is 1, 2, 3 or 4: When N is 1, $\overline{I_1} = 1\ pkt, q_1 = \frac{1}{2}$. Let $T_1 = T$, where T is the service time to transmit a packet over one hop when the source and destination nodes are one hop away.

When N is 2, each TCP packet travels two wireless links to reach the destination. Therefore, approximately $T_2 = 2T$. During the transmission from node 0 to node 1, and then from node 1 to node 2, only 1 TCP packet can be transmitted in the system without collision. Therefore $\overline{I_2} = 1\ pkt$. In the *string* topology, the source node only forwards TCP packets, the destination node only forwards ACK packets but the middle node forwards both TCP and ACK packets. Since one ACK packet is assumed for each TCP packet, the middle node sends out packets twice as much as the source and destination node. As a result, the average probability of the middle node to access the channel successfully is twice as much as that of the source and destination node. In addition, the summation of the average probability of all nodes to access the channel is 1. Derivation from these relationships gives $q_2 = \frac{1}{4}$. When N is 3, by similar analysis we have $\overline{I_3} = 1\ pkt, T_3 = 3T, q_3 = \frac{1}{6}$.

When N is 4, it looks that node 0 and node 3 could send packets concurrently without collision. However, although node 3 is outside of node 1's transmission range, it is within the carrier sensing range and interference range of node1. Node 3 is thus a potential hidden terminal of the transmission pair node 0 to node 3. Consequently, only 1 TCP packet can be transmitted in the system without collision. This analysis is consistent with that in [4]. Further derivation gives $\overline{I_4} = 1\ pkt, T_4 = 4T, q_4 = \frac{1}{8}$.

2) N is 5, 6, 7, or 8: When N is 5, each TCP packet crosses five wireless hops to reach the destination. Among the five hops from node 0 to node 5, the 1^{st} hop (i.e., from node 0 to node 1) and the 5^{th} hop (i.e., from node 4 to node 5) can be utilized at the same time as shown in Fig. 5. But when the 2^{nd}, 3^{rd} or 4^{th} hop is used, only that hop can be used without collision.

Assuming that the system is utilized fully, the number of packets in the system is:

$$I_5 = \begin{cases} 2\ pkts & \text{when the } 1^{st} \text{ and } 5^{th} \text{ hops are occupied} \\ 1\ pkt & \text{when the } 2^{nd}, 3^{rd} \text{ or } 4^{th} \text{ hop is occupied} \end{cases}$$

Since each hop has equal opportunity to be utilized, the average number of packets in the system is: $\overline{I_5} = 2\times\frac{2}{5}+1\times\frac{3}{5}\ pkts$.

Although there are six nodes, however, node 0 competes to use the channel locally with nodes 1, 2, 3 and 4 only. The probability for node 0 to access the channel can therefore be taken as the same as when there are 4 hops. Thus $q_5 = \frac{1}{8}, T_5 = 5T$.

When N is 6, each TCP packet crosses six wireless links

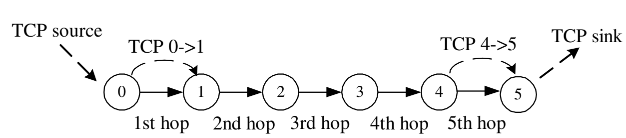

Fig. 5. Two TCP packets are transmitted together when $N = 5$.

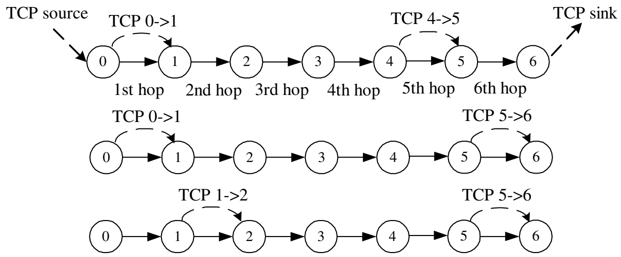

Fig. 6. Two TCP packets are transmitted together when $N = 6$.

to reach the destination. Fig. 6 shows the scenarios when two of the six wireless links are utilized at the same time. The number of packets in the system is:

$$I_6 = \begin{cases} 2\ pkts & \text{when the } (1^{st} \text{ or } 2^{nd}) \text{ and } (5^{th} \text{ or } 6^{th}) \\ & \text{hops are occupied} \\ 1\ pkt & \text{when the } 3^{rd} \text{ or } 4^{th} \text{ hop is occupied} \end{cases}$$

The average number of packets in the system is: $\overline{I_6} = 2 \times \frac{4}{6} + 1 \times \frac{2}{6}\ pkts$. It is also easy to get $q_6 = \frac{1}{8}, T_6 = 6T$.

Similarly, when N is 7, we get

$$I_7 = \begin{cases} 2\ pkts & \text{when the } (1^{st}, 2^{nd} \text{ or } 3^{rd}) \text{ and} \\ & (5^{th}, 6^{th} \text{ or } 7^{th}) \text{ hops are occupied} \\ 1\ pkt & \text{when the } 4^{th} \text{ hop is occupied} \end{cases}$$

$\overline{I_7} = 2 \times \frac{6}{7} + 1 \times \frac{1}{7}\ pkts, q_7 = \frac{1}{8}, T_7 = 7T.$

When N is 8, $I_8 = \overline{I_8} = 2\ pkts$, when the ($1^{st}$, 2^{nd},3^{rd} or 4^{th}) and (5^{th}, 6^{th}, 7^{th} or 8^{th}) hops are occupied, and $q_8 = \frac{1}{8}, T_8 = 8T$.

3) Generalization of TCP Service Rate: Let $N = 4M + j$, where $M \geq 1, j \subseteq \{1, 2, 3, 4\}$, the generalization of the above analysis is as follows.

When N is $4M$, we have $I_{4M} = \overline{I_{4M}} = M\ pkts$.

When N is $4M + 1$,

$$I_{4M+1} = \begin{cases} M+1\ pkts & \text{when the } 1^{st} \text{ and } 5^{th} \text{ and ...} \\ & (4M+1)^{th} \text{ hops are occupied} \\ M\ pkts & \text{when the } (2^{nd}, 3^{rd} \text{ or } 4^{th}) \text{ and} \\ & (6^{th}, 7^{th} \text{ or } 8^{th}) \text{ and ...} \\ & ((4M-2)^{th}, (4M-1)^{th} \text{ or} \\ & (4M)^{th}) \text{ hops are occupied} \end{cases}$$

$\overline{I_{4M+1}} = (M+1) \times \frac{M+1}{4M+1} + M \times \frac{(4M+1)-(M+1)}{4M+1}\ pkts.$

Similar derivation gives:

$\overline{I_{4M+2}} = (M+1) \times \frac{2(M+1)}{4M+2} + M \times \frac{(4M+2)-2(M+1)}{4M+2}\ pkts.$

$\overline{I_{4M+3}} = (M+1) \times \frac{3(M+1)}{4M+3} + M \times \frac{(4M+3)-3(M+1)}{4M+3}\ pkts.$

Finally, it is summarized that:

$$\overline{I_{4M+j}} = (M+1) \times \frac{j(M+1)}{4M+j} + M \times \frac{(4M+j)-j(M+1)}{4M+j} \qquad (9)$$

where $M \geq 1, j \subseteq \{1, 2, 3, 4\}$.

Combining (9) and the analysis when N is 1, 2, 3 or 4, and we have the average number of packets in the system as:

$$\overline{I_{4M+j}} = \frac{j(M+1)^2 + (4-j)M^2}{4M+j} \qquad (10)$$

where $M \geq 0, j \subseteq \{1, 2, 3, 4\}$.

The service time to transmit a packet is generalized as $T_{4M+j} = (4M + j) \times T$. From (8), the TCP service rate is derived as:

$$U_{t(4M+j)} = \frac{j(M+1)^2 + (4-j)M^2}{(4M+j)^2} \times \frac{1}{T} \qquad (11)$$

where $M \geq 0, j \subseteq \{1, 2, 3, 4\}$.

The average probability of the source to access the channel is:

$$q_{4M+j} = \begin{cases} \frac{1}{2j} & \text{when } M = 0 \text{ and } j \subseteq \{1, 2, 3, 4\} \\ \frac{1}{8} & \text{when } M \geq 1 \text{ and } j \subseteq \{1, 2, 3, 4\} \end{cases} \qquad (12)$$

C. Generalisationof TCP Throughput

Substituting (11) and (12) into (7), the TCP throughput becomes:

$$\lambda_{4M+j} = \begin{cases} \frac{1}{j(1+(2j-1)\rho)} \times \frac{1}{T} \\ \qquad \text{when } M = 0 \text{ and } j \subseteq \{1, 2, 3, 4\} \\ \frac{j(M+1)^2+(4-j)M^2}{(4M+j)^2} \times \frac{1}{1+7\rho} \times \frac{1}{T} \\ \qquad \text{when } M \geq 1 \text{ and } j \subseteq \{1, 2, 3, 4\} \end{cases} \qquad (13)$$

When M goes to infinity, $N = 4M + j$ goes to infinity, we have

$$\lim_{M\to\infty} \lambda_{4M+j} = \frac{1}{4} \times \frac{1}{T} \times \frac{1}{1+7\rho} \qquad (14)$$

Equation (13) shows that the TCP throughput is independent of the maximum TCP congestion window size W; instead it is decided by (a) $4M + j$, the value of the number of hops of the *string* topology, (b) ρ, the ratio of service rate of TCP packet to ACK packet, and (c) T, the time needed for a TCP packet to be transmitted over a single hop. Furthermore, (14) illustrates that as the number of hops increases and goes to infinity, TCP throughput converges to a constant value.

V. Simulation and Numerical Results

The analytical results are verified by the comparision of simulation results in *ns2* and numerial results in *matlab*.

In simulations, $N + 1$ nodes form a *string* (see Fig. 1) with adjacent nodes 200 *m* apart. All nodes communicate with identical, half-duplex wireless radios which have a bandwidth of 2 *Mbps* and a nominal transmission radius of 250 *m*. Nodes have carrier sense radius of 550 *m* and interference radius of 550 *m*. They are configured with the Dynamic Source Routing (DSR) protocol. One TCP Reno session is introduced from node 0 (TCP source) to node N (TCP destination) to transfer FTP bulk data, crossing N hops. TCP and ACK packets size are of 1460 bytes and 40 bytes respectively. Simulations were run with various numbers of hops and various maximum TCP congestion window sizes. One simulation with the same number of hops and the same maximum TCP congestion window size was run for five times, each for 300 *secs*, and the overall throughput was measured from 50 *secs* to 250 *secs*. During the time the throughput is quite stable, but the average of five simulations was used as the simulation result.

To get the numerical results in *matlab*, two parameters are needed:

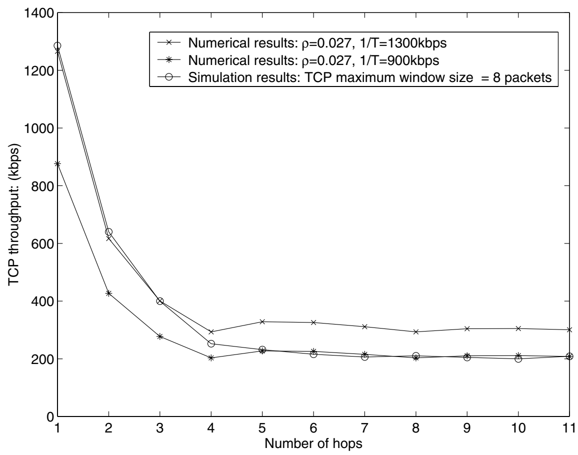

Fig. 7. Comparison of simulation and numerical results: TCP throughput with varying number of hops.

- ρ: the ratio of service rate of TCP packet to ACK packet. Here $\rho = \frac{40\ bytes}{1460\ bytes} = 0.027$, i.e., the ratio of ACK packet size to TCP packet size.
- $\frac{1}{T}$: the average transmission rate of one packet over a single hop link, where T is defined as the average transmission time of one packet over one hop link. However, $\frac{1}{T}$ cannot be easily assigned a value considering the bandwidth used on the channel contention, MAC control packets exchange, etc. Instead, we decide the value of $\frac{1}{T}$ with the help of simulation results so that the numarical results could best match the simulation results.

Fig. 7 shows the comparison of simulation and numerical results of the TCP throughput as the number of hops changes from 1 to 11. Simulation results are presented when the maximum TCP congestion window size is 8 packets. Numerical results are presented with two values of $\frac{1}{T}$, so there are three curves in Fig. 7: the simulation results, the numerical results when $\frac{1}{T} = 1300$ *kbps*, and the numerical results when $\frac{1}{T} = 900$ *kbps*.

It is observed that the numerical results with $\frac{1}{T} = 1300$ *kbps* match the simulation results well at $N = 1, 2, 3$ or 4 hops, and the numerical results with $\frac{1}{T} = 900$ *kbps* match the simulation results well at $N \geq 5$. The reason is as follows. When $N = 1, 2, 3$ or 4, no link is used simultaneously due to the hidden terminal problem. When $N \geq 5$, there are links used simultaneously because of channel spatial resue. However, although channel spatial reuse helps improve the overall system utilization, it has higher local contention overhead. This results in a lower payload transmission rate. Therefore, $\frac{1}{T}$ is chosen as 900 *kbps* when $N \geq 5$ and 1300 *kbps* when $N = 1, 2, 3$ or 4, respectively. In Fig. 7, TCP throughput is shown to decrease rapidly when the number of hops increases from 1, and stabilizes when the number of hops becomes large. This is in line with the observation we described in the introduction and verifies the analysis in (13) and (14).

Simulation results (not presented here due to space constraints) also show that when the number of hops is fixed, TCP throughput is kept constant independent of the maximum TCP congestion window size. This is consistent with (13) where the TCP throughput (λ) is not a function of the maximum TCP congestion window size (W). This result, however, looks inconsistent with the claim in [4] that "given a specific network topology and flow patterns, there exists a TCP window size W^*, at which TCP achieves best throughput via improved spatial channel reuse". However, a close look at Fig. 2 (a) and (b) in [4] reveals that TCP throughput at the "optimal" TCP window size is not much higher than those at non-optimal TCP window sizes; furthermore, TCP throughput is a constant at most values of TCP window size. This observation is consistent with our analysis here.

VI. Conclusion

This study takes a step to theoretically analyze TCP performance in adhoc wireless networks. The analysis is based on a *string* topology containing N hops. A system model for analyzing TCP performance in adhoc wireless networks is proposed, which considers packet buffering, nodes' contention for access to wireless channel and spatial reuse of wireless channel. Markov chain modelling is applied to analyse the system model. Analytical results are presented to show that when the number of hops that the TCP session crosses is fixed, the TCP throughput is independent of the TCP congestion window size. When the number of hops increases from one, the TCP throughput decreases first, and then stabilizes when the number of hops becomes large. The analysis is validated by comparing the numerical results with simulation results.

References

[1] D. Bertsekas and R. Gallager. *Data Networks*. Prentice Hall, second edition, 1992.

[2] Vaduvur Bharghavan, Alan J. Demers, Scott Shenker, and Lixia Zhang. MACAW: A media access protocol for wireless LAN's. In *SIGCOMM*, pages 212–225, 1994.

[3] H. M. Chaskar, T. V. Lakshman, and U. Madhow. TCP over wireless with link level error control: analysis and design methodology. *IEEE/ACM Transactions on Networking*, 7(5):605–615, October 1999.

[4] Z. Fu, H. Luo, P. Zerfos, S. Lu, L. Zhang, and M. Gerla. The impact of multihop wireless channel on TCP performance. *IEEE Transactions on Mobile Computing*, 4(2):209–221, March 2005.

[5] M. Gerla, R. Bagrodia, L. Zhang, K. Tang, and L. Wang. TCP over wireless multi-hop protocols: simulation and experiments. In *Proceedings of IEEE ICC'99*, pages 1089–1094, Vancouver, USA, June 1999.

[6] G. Holland and N. H. Vaidya. Analysis of TCP performance over mobile ad hoc networks. In *Proceedings of IEEE/ACM MOBICOM'99*, pages 219–230, Seattle, WA, USA, August 1999.

[7] T. V. Lakshman, U. Madhow, and B. Suter. TCP/IP performance with random loss and bidirectional congestion. *IEEE/ACM Transactions on Networking*, 8(5):541–555, October 2000.

[8] H. Xiao, K. C. Chua, K. G. Seah, and A. Lo. A quantitative analysis of TCP performance over wireless multihop networks. In *Proceedings of IEEE VTC2001-fall*, pages 7–11, May 2001.

[9] H. Xiao, K. G. Seah, A. Lo, and K. C. Chua. On service differentiation in multihop wireless networks. In *ITC Specialist Seminar on Mobile Systems and Mobility*, pages 1–12, Lillehammer, Norway, March 2000.