TCP over Wireless Multi-hop Protocols: Simulation and Experiments

Mario Gerla, Rajive Bagrodia, Lixia Zhang, Ken Tang, Lan Wang
{gerla, rajive, lixia, ktang, lanw}@cs.ucla.edu

Wireless Adaptive Mobility Laboratory
Computer Science Department
University of California, Los Angeles
Los Angeles, CA 90095
http://www.cs.ucla.edu/NRL/wireless

Abstract

In this study we investigate the interaction between TCP and MAC layer in a wireless multi-hop network. This type of network has traditionally found applications in the military (automated battlefield), law enforcement (search and rescue) and disaster recovery (flood, earthquake), where there is no fixed wired infrastructure. More recently, wireless "ad-hoc" multi-hop networks have been proposed for nomadic computing applications. Key requirements in all the above applications are reliable data transfer and congestion control, features that are generally supported by TCP. Unfortunately, TCP performs on wireless in a much less predictable way than on wired protocols.

Using simulation, we provide new insight into two critical problems of TCP over wireless multi-hop. The first is the conflict between data packets and ACKs, which causes TCP performance to degrade for window sizes greater than 1 packet. The second is the interaction between MAC and TCP layer backoff timers which causes severe unfairness and capture conditions. In the paper, we identify these problems in several representative simulation runs on various topologies and traffic patterns and indicate possible remedies to improve TCP efficiency over a wireless multi-hop network.

1. Introduction

The rapid advancement in portable computing platforms and wireless communication technology has led to significant interest in the design and development of protocols for instantly deployable, wireless networks often referred to as "Ad-Hoc Networks". Ad-hoc networks are required in situations where a fixed communication infrastructure, wired or wireless, does not exist or has been destroyed. The applications span several different sectors of society. In the civilian environment, they can be used to interconnect workgroups moving in an urban or rural area or a campus and engaged in collaborative operation such as distributed scientific experiments and search and rescue. In the law enforcement sector, applications such as crowd control and border patrol come to mind. In the military arena, the modern communications in a battlefield theater require a very sophisticated instant infrastructure with far more complex requirements and constraints than the civilian applications [8].

In a nutshell, the key characteristics which make the design and evaluation of ad-hoc networks unique and challenging include mobility, unpredictable wireless channel such as fading, interference and obstacles, broadcast medium shared by multiple users and very large number of heterogeneous nodes (e.g., thousands of sensors).

To these challenging physical characteristics of the ad-hoc network, we must add the extremely demanding requirements posed on the network by the typical applications. These include multimedia support, multicast and multi-hop communications. Multimedia (voice, video and image) is a must when several individuals are collaborating in critical applications with real time constraints. Multicasting is a natural extension of the multimedia requirement. Multi-hopping is justified (among other things) by the limited power of the mobile devices, by obstacles and by the desire to reuse frequency and/or code.

Two key requirements of the ad-hoc network environment are reliable data transfer and congestion control. These features are generally supported by TCP. An important question is how TCP (which has been designed and fine-tuned for wired networks) interacts with the wireless protocols, in particular the MAC layer. Both MAC and TCP layers strive to provide efficient transport in a shared environment, with some degree of efficiency and with protection from errors and interference. The MAC layer however has only a myopic view of the network, which is a critical limitation in multi-hop networks. In contrast, TCP provides a true end-to-end control on errors and congestion.

In this paper, we study the TCP/MAC layer interaction via simulation. The simulation platform used is GloMoSim [18]. GloMoSim is a parallel simulation environment implemented in PARSEC, PARallel Simulation Environment for Complex Systems [1]. It includes several wireless protocols in its library (radio propagation, mobility, MAC, network, transport and applications). Most importantly, GloMoSim permits the detailed modeling of several layers and the study of their interaction, yet preserving very good runtime efficiency and yielding manageable execution time.

The rest of the paper is organized as follows: Section 2 reports the configuration and parameters we used for our simulation. TCP over the MAC layer experiments are examined in section 3. Section 4 summarizes our grid topology simulation results. Finally, section 5 concludes the paper.

2. Experimental Configuration and Parameters

We consider three different types of topologies: a string topology with 8 nodes (0 through 7) as shown in Fig. 1; a ring topology with 8 nodes as shown in Fig. 2; and a grid topology with up to 100 nodes as shown in Fig. 3. Radio channels are bidirectional; the arrows indicate the direction of data packet transmissions. ACKs travel in the opposite direction. The distance between two neighbor nodes is equal to the radio transmission range. TCP or UDP connections are established between different node pairs. These connections carry large file transfers (i.e. infinite backlog). In some cases (grid topology), they carry interactive traffic. Nodes are static (no mobility). Free space channel model is used in the simulation model. Perfect channel is assumed (no external noise). Channel data rate is 2Mbps.

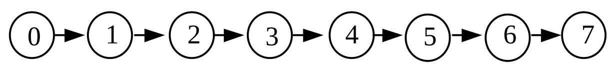


Fig. 1. String Topology.

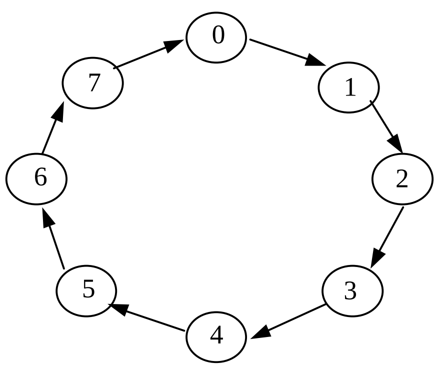


Fig. 2. Ring Topology.

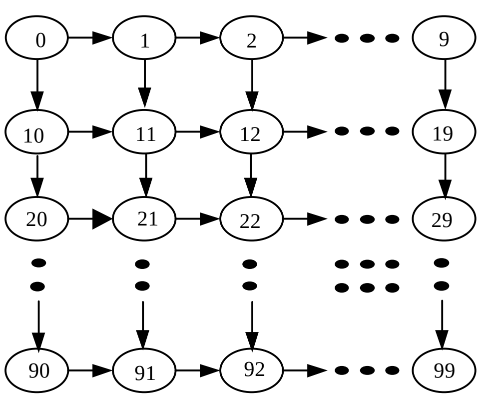


Fig. 3. Grid Topology.

Two MAC protocols are considered: CSMA and FAMA. These protocols were chosen because they are representative of a broad class of MAC schemes used in wireless LANs. CSMA (Carrier Sense Multiple Access) requires carrier sensing before transmission. If the channel is free, the packet is transmitted immediately. Otherwise, it is rescheduled after a random timeout. The major limitation of CSMA is the “hidden” terminal problem. Nodes 2 and 4 (in Fig. 1) cannot hear each other but are both within reach of node 3. They may transmit simultaneous (in spite of carrier sensing) and thus cause a collision at node 3. CSMA was used first in the Packet Radio network in the mid 1970’s [10]. It is also used in several popular wireless LANs. FAMA (Floor Acquisition Multiple Access) uses the RTS (Request To Send) and CTS (Clear To Send) exchange to prepare the floor for data transmission (thus avoiding "hidden terminal" collision in most cases) [7]. FAMA is an experimental MAC protocol specifically developed for the Glomo DARPA program. It bears close resemblance to the IEEE 802.11 protocol, which is being proposed as the standard of wireless LANs as well as ad-hoc wireless networks [12].

Each node has a 25 packet MAC layer buffer pool. Incoming packets that find the buffer full are dropped. Scheduling of packet transmissions is FIFO. The routing protocol is Distance Vector (Bellman-Ford). In order to avoid interference between routing update packets and TCP packets, the DV algorithm is run only at the start of each experiment to initialize the routing tables.

The TCP simulation model is an accurate replica of the TCP code running in the Internet hosts today. The TCP simulation code was generated from FreeBSD 2.2.4 code. In particular, window size grows progressively until it reaches the advertised window or until packet loss is detected. In the latter cases, window size is halved (fast retransmission and fast recovery) or abruptly reduced to 1 (slow start). In our simulation, we can "force" the maximum TCP window to be at a certain value by setting the advertised window to such value (e.g., 1460B). TCP packet length is assumed fixed at 1460B. In some experiments, packet loss due to channel interference is so high that some TCP connections are timed out and closed. To allow the simulator to run to completion, we artificially increased the maximum number of retransmissions, thus avoiding premature TCP connection closure.

3. TCP over MAC Layer

Previous MAC layer simulation experiments at UCLA have uncovered two major weaknesses of the MAC layer: (a) CSMA and, to a lesser extent, FAMA suffer from the hidden terminal losses; (b) one or more stations tend to “capture” the channel in heavy load situation. Thus, in the first place loss recovery must be provided by an upper layer, either by the link or transport layer. In this study, we consider loss recovery via the TCP transport layer protocol, as it is the most popular solution in wireless LANs and it is the easiest to implement in a multi-hop wireless environment. Secondly, we wish to remove capture. The question here is whether TCP improves the capture situation, or makes it worse.

We start with a single TCP connection that covers a variable number of hops, from 1 to 7 hops. In the first set of experiments, TCP window (W) is 1460B. Thus, W = 1 packet. The results for CSMA and FAMA throughputs as a function of number of hops H are reported in Table 1. One can verify that throughput values match exactly the analytic predictions for a send-and-wait protocol. The throughput is inversely proportional to the hop distance. CSMA throughput is slightly higher than FAMA [17] because of RTS/CTS overhead in the latter.

Number of Hops	CSMA	FAMA
1	1838.4	1476.5
2	921.3	718.7
3	614.8	475.4
4	461.4	355.3
5	369.2	287.5
6	307.7	239.1
7	263.4	204.7

Table 1. Throughput (Kbps), Single TCP Connection, Variable Number of Hops, W = 1460B.

Next, we set W = 32KB. Here, the TCP protocol dynamically adjusts the congestion window as required. As window is increased, multiple packets and multiple ACKs travel on the path in opposite directions, creating interference and collisions (both in CSMA and FAMA). We would expect that in balance the window increase improves performance since for 7 hops, for example, analysis shows that the optimal throughput (assuming optimal scheduling of packet and ACK transmissions along the path) is achieved for W = 3 x 1460B. The simulation results in Table 2 indicate otherwise. CSMA throughput collapses when $H \geq 3$. FAMA does slightly better than CSMA, yet with throughput much lower than with W = 1460B.

Number of Hops	CSMA	FAMA
1	1791.2	1458.7
2	439.5	716.2
3	0.5	389.4
4	0.5	71.4
5	0.5	13.5
6	0.5	72.8
7	0.5	66.9

Table 2. Throughput (Kbps), Single TCP Connection, Variable Number of Hops, W = 32KB.

From the above results we conclude that there is no gain in using W larger than single packet size even on connections covering multiple hops.

Next, we consider the ring topology in Fig. 2. The 8 nodes are engaged in single hop file transfer connections (0-1, 1-2, etc). We run both CSMA and FAMA. The results are reported in Table 3. We only consider W = 1460B since we have observed that W > 1460B typically degrades performance in the multi-hop environment. The key performance measures of interest are throughput efficiency and fairness.

TCP Connection	CSMA	FAMA
0-1	275.0	34.9
1-2	298.7	34.7
2-3	180.5	28.4
3-4	400.8	0.1
4-5	426.5	1472.6
5-6	253.6	1.2
6-7	307.1	0.2
7-0	362.8	1506.9
Total Throughput	2505.0	3079.0

Table 3. Throughput (Kbps) in Ring Topology, W = 1460B.

We start with CSMA and note that with TCP the protocol is reasonably fair. Aggregate throughput is 2.5Mbps, which is quite good considering the fact that the maximum theoretical throughput achievable on the ring is 4Mbps (i.e., two simultaneous session at least 3 hop apart).

With FAMA, TCP yields 3Mbps! This compares very favorably with the theoretical maximum of 4Mbps. The solution is quite unfair, however. Two links capture the entire throughput. The remaining links have small to negligible throughput.

Finally, we consider the case of a TCP connection spanning several hops sharing the path with several single hop connections. The experimental configuration (see Fig. 4) consists of a string with 8 nodes, seven single hop connections (0-1, 1-2, etc) and one multi-hop connection (0-7). The results are reported in Table 4. We find zero or negligible throughput for the (0-7) connection for both CSMA and FAMA. This is not unexpected since, without link level ACKs the probability of a packet making it through 7 hops is very slim!

TCP Connection	CSMA	FAMA
0-1	358.7	334.2
1-2	20.6	437.5
2-3	811.3	336.9
3-4	960.7	147.0
4-5	21.8	834.0
5-6	0.0	248.9
6-7	1630.1	387.7
7-0	0.0	0.1
Total Throughput	3803.3	2726.3

Table 4. Throughput (Kbps) in String Topology with 0-7 Data Stream, W = 1460B.

In the FAMA experiments also reported in Table 4, again, we cannot get any significant throughput from 0 to 7. The behavior of the single hop sources is fairer than in CSMA, but aggregate throughput is lower (2.7Mbps).

In an attempt to favor the 0-7 connection, we did increase its window to 32KB. This however had no positive effects, yielding results similar to those of Table 2. In a separate set of experiments, we reduced the length of the multi-hop connection progressively from 7 hops (i.e., 0-7) down to 2 hops (i.e., 2-4). We were able to observe significant traffic (111Kbps) only in the 2-4 case with CSMA. Zero throughput was yielded for 2-4 by FAMA.

It is interesting to compare the behavior of the single hop connections in the string with the ring topology. Fairness is much worse in the string than in the ring. Some nodes seem to capture the channel while others are locked out. Aggregate throughput is better in the string than in the ring (3.8Mbps vs. 2.5Mbps). This is expected since the maximum theoretical throughput on the string (based on optimal scheduling) is 6Mbps, versus 4Mbps on the ring topology.

In summary, the following lessons were learned from the TCP experiments in heavy file transfer load. First of all, a large window has a negative effect especially on CSMA. A window of one packet size provides by far the best results. Secondly, capture is not removed. In fact, it is often made worse by TCP. Apparently, the backoffs in MAC and TCP reinforce each other, emphasizing capture and unfairness. Finally, unfairness is particularly severe with respect to multi-hop connections. No traffic gets through beyond two hops when the network is heavily loaded.

4. Grid Topology

The previous experiments have been based on linear topologies (string or ring) with just one type of traffic (file transfers). In this section, we wish to model a more realistic environment. Thus, we select a grid topology where each node has four neighbors except for the nodes positioned at the edge of the grid. Network size varies from 4 to 100 nodes. The 10 X 10 grid is shown in Fig. 3. Two types of traffic are injected in this network: FTP traffic along all vertical paths (i.e., from 0 to 90, from 1 to 91, etc), and interactive traffic along all horizontal hops (i.e., from 0 to 1, from 1 to 2, from 2 to 3, … from 10 to 11, etc). The interactive traffic is modeled by a constant offered rate of fixed size packets. The rate is uniform over the network and varies from experiment to experiment. File transfer sources have constant, heavy supply of packets to transmit.

The performance measure of interest is again throughput. The key variables are offered interactive load and network size. We ran experiments with the following loads (per Interactive TCP connections): 1.5, 7.7, 23.3 and 233.6Kbps; and with the following grid sizes: 2 X 2, 6 X 6, 7 X 7 and 10 X 10. We report below detailed results for the 10 X 10 experiment with 3 loading conditions: 1.5Kbps, 23.3Kbps and 233.6Kbps. Summary results will be provided for the other cases.

We begin with the analysis of the CSMA experiment on the 10 X 10 grid with offered load of 1.5Kbps. The interactive sources achieve their full throughput, i.e., they manage to discharge all packets that arrive. The FTP sources fare less well (see Fig. 4). They achieve throughputs ranging from 12Kbps to 64Kbps, with average of 31.9Kbps. The theoretical maximum is 200Kbps (9 hops, W = 1 packet). The gap between theoretical maximum and measured values is due to interference from neighbor FTP connections as well as from interactive traffic. Particularly damaging to multi-hop FTP connections is the high loss rate on the links and the lack of link loss recovery. The FTP throughput at the borders of the grid (i.e., 0 to 90 and 9 to 99) is higher than in the interior since the interference is lower there. Unfairness due to capture is already quite evident, even at this low level of interactive traffic.

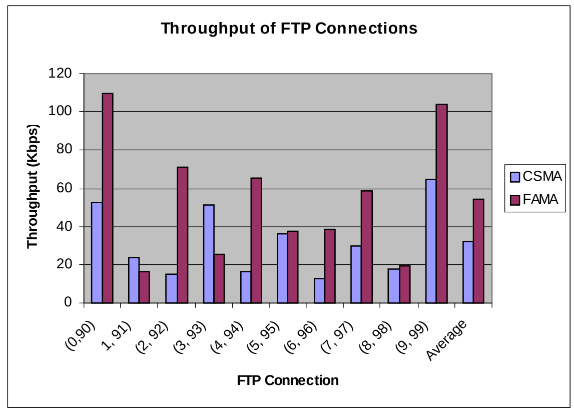


Fig. 4. 10 X 10 Grid Topology, Offered Interactive Load of 1.5Kbps.

Next, we shift our attention to FAMA for the same interactive load (see Fig. 4). Interactive connection performance is about the same as with CSMA. FTP average FAMA throughput is higher than CSMA, 54.5Kbps as compared with 31.9Kbps in CSMA. FAMA is better protected against hidden terminal losses. This reduces FAMA losses and improves its throughput. Unfairness and capture, however, are present also in FAMA (maximum of 109Kbps, minimum of 15Kbps). Again, unfairness is due to the progressively increasing random timeouts. As the interactive offered load is increased to 23.3Kbps (see Fig. 5), we start noticing losses in the interactive throughput as well. Namely, the interactive sources cannot keep up with the offered load. In the CSMA experiment the interactive throughput is 20.7Kbps. In FAMA, it is 21.4Kbps. As for the FTP connections, the increase in interfering load causes strong degradation and strong capture, especially for CSMA. Maximum and minimum values in CSMA are 6.1Kbps and 0.04Kbps. Obviously, some connections are practically locked out! FAMA fares a bit better, with maximum and minimum of 22.9Kbps and 1.9Kbps, respectively.

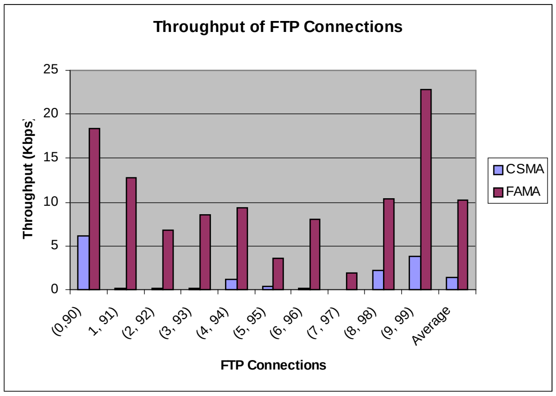


Fig. 5. 10 X 10 Grid Topology, Offered Interactive Load of 23.3Kbps.

For interactive load of 233.6Kbps (see Fig. 6), the interactive sources show strong signs of capture/unfairness. In CSMA we measure a maximum of 121Kbps and minimum of 11.4Kbps for interactive throughput. With FAMA, maximum and minimum are 180.5Kbps and 36Kbps. Average interactive connection throughput is 53Kbps in CSMA and 73.8Kbps in FAMA. As for the FTP throughput, this virtually collapses at this load. The average FTP throughput is 0.087Kbps, surprisingly the same for CSMA and FAMA!

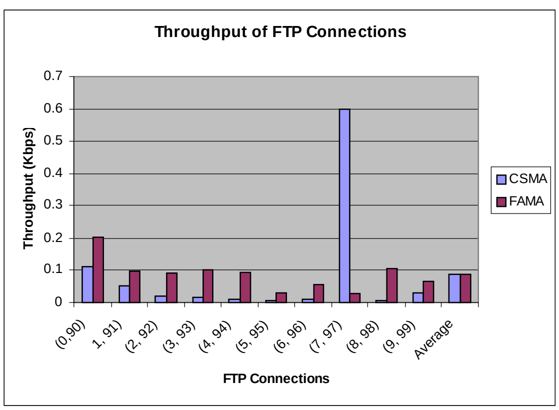
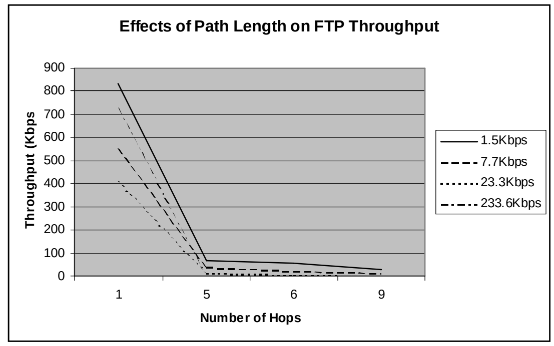
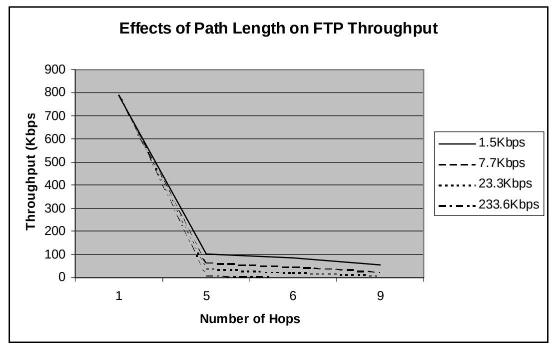


Fig. 6. 10 X 10 Grid Topology, Offered Interactive Load of 233.6Kbps.

Fig. 7(a) and 7(b) show average CSMA and FAMA FTP throughput, respectively, as a function of path length for various values of interactive load. These results confirm our earlier findings that end-to-end TCP throughput drops very rapidly as path length increases. The drop is caused by link loss, which in turn is caused by interference from neighboring FTP connections and from interactive traffic.



(a) CSMA



(b) FAMA

Fig. 7. Average FTP Throughput vs. Number of Hops.

More precisely, if the packet loss probability on a link is p (assumed the same for data packets and ACKs) and hop length is h, then one finds that the TCP throughput is given by:

$$R = R_0 \left[2^{-\left(\frac{1}{(1-p)^{2h}}-1\right)} \right]$$

where R_0 = throughput in absence of packet loss. Thus, throughput R drops faster than exponentially with hop length. The more than exponential drop derives from two facts: (a) the probability that both packet and ACK survive is $(1-p)^{2h}$, and; (b) the TCP retransmission time-out doubles after each retransmission, thus average timeout is:

$$T_{out} = T_0 \left[2^{\frac{1}{(1-p)^{2h}}-1} \right]$$

In summary, the grid experiments confirm the irregular behavior observed in linear topologies. Namely, end-to-end throughput decays exponentially with hop length. FAMA is superior to CSMA (in terms of throughput). Yet both FAMA and CSMA suffer of unfairness problems (mainly due to capture). New insights were also offered by these grid experiments. For example, the FTP degradation depends critically on the level of single hop interfering traffic. It is less sensitive, on the other hand, to the interference caused by other multi-hop TCP connections.

5. Conclusion

The focus of the paper has been the MAC/TCP layers interaction in a multi-hop radio network. We have considered two representative MAC layers - CSMA and FAMA. The main findings of the study are:

(a) Both CSMA and FAMA exhibit capture under TCP. Namely, connections take turns in "capturing" the channel. The capture may last several seconds, even minutes. This is clearly unacceptable in real time environments such as the battlefield.
(b) Multi-hop TCP connections are at clear disadvantage with respect to single hop connections. In case of channel contention, a multi-hop connection gets zero throughput in most cases. Moreover, the best performance is achieved with W = 1 packet (to avoid conflict between multiple packets and outstanding ACKs).
(c) FAMA yields overall better throughput than CSMA. It is however more prone to capture than CSMA.
(d) Capture as well as other layer interactions can be adequately studied only using a detailed simulation platform like GloMoSim. Simulators which abstract some of the layer features (e.g., ignoring retransmission time out policies) may just miss the capture behavior.

The results indicate that more research is necessary to make TCP and MAC layers work well together in a multi-hop environment. More precisely:

(a) MAC timeouts must be revisited, adding for instance "vacations" after success;
(b) Link level ACKs must be introduced in order to reduce link loss rates and make TCP work efficiently over multi-hop paths;
(c) End-to-end cumulative ACKs, with selective retransmit feature, should be considered to make TCP more efficient in heavy collision environment;
(d) Finally, priorities and scheduling should be designed to maintain fairness and to support QoS connections.

References

[1] R. Bagrodia, R. Meyer, et al., “PARSEC: A Parallel Simulation Environment for Complex System”, to appear in Computer Magazine, 1998.

[2] Bikram Bakshi, Krishna, P., Pradhan, D.K., and Vaidya, N.H., “Performance of TCP over Wireless Networks”, 17th Intl. Conf. on Distributed Computing Systems (ICDCS), Baltimore, May, 1997.

[3] Hari Balakrishnan, Srinivasan Seshan, Randy H. Katz, “Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks”, ACM Wireless Networks, 1(4), December 1995.

[4] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LAN' s," ACM SIGCOMM, 1994.

[5] C. D. Carothers, Richard M. Fujimoto, Yi-Bing Lin and Paul England, “Distributed Simulation of Large-scale PCS Networks, MASCOTS ' 94. Proceedings of the Second International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (Cat. No.94TH0621-3), MASCOTS ' 94.

[6] J.G. Clearly, J. J. Tsai, “Conservative Parallel Simulation of ATM Networks”, Proceedings of PADS 1996.[Floyd94] S. Floyd, “TCP and Explicit Congestion Notification”, ACM Computer Communication Review, V. 24 N. 5, October 1994, p. 10-23.

[7] C. Fullmer and J.J. Garcia-Luna-Aceves, “Floor Acquisition Multiple Access (FAMA) for packet radio networks”, Computer Communication Review, vol. 25, (no. 4), (ACM SIGCOMM ' 95, Cambridge, MA, USA, 28 Aug.-1 Sept. 1995.) ACM, Oct. 1995.

[8] M. Gerla and J.T.-C. Tsai, “Multicluster, mobile, multimedia radio network”, ACM/Baltzer Journal of Wireless Networks. vol. 1, (no. 3), 1995, p. 255-265.

[9] T. Holvoet and P. Verbaeten, “Using Agents for Simulating and Implementing Petri nets”, Proceedings of PADS 1997.

[10] John Jubin and Janet D. Tornow, “The DARPA Packet Radio Network Protocols”, Proceedings of the IEEE, Jan. 1987.

[11] P. Karn, “MACA –a New Channel Access Method for Packet Radio”, in ARRL/CRRL Amateur radio 9th Computer Networking Conference, ARRL, 1990.

[12] C. R. Lin and M. Gerla, “MACA/PR: An asynchronous multimedia multi-hop wireless network”, IEEE INFOCOMM, 1997.

[13] P. Martini, M. Rumekasten, J. Tolle, “Tolerant Synchronization for Distributed Simulations of Interconnected Computer Networks”, Proceedings of PADS 1997.

[14] Robert Morris, “TCP Behavior with Many Flows”, International Conference on Network Protocols, October 1997.

[15] T. Rappaport and S. Y. Seidel, SIRCIM: Simulation of Indoor Radio Channel Impulse Response Models, VTIP, Inc., 1990.

[16] J. Short, R. Bagrodia, and L. Kleinrock, "Mobile Wireless Network System Simulation", Proceedings of ACM Mobile Communication Networking Conference -- Mobicom ' 95, November 1995.

[17] Andrew S. Tanenbaum, “Computer Networks: Third Edition”, Prentice Hall PTR, New Jersey, 1996.

[18] X. Zeng, R. Bagrodia and M. Gerla, “GloMoSim: a Library for the Parallel Simulation of Large-scale Wireless Networks”, will appear in Proceedings of PADS’1998.