

What does using TCP as an Evaluation Tool Reveal about MANET Routing Protocols?*

Sundaram Rajagopalan

Department of Computer and Info. Sciences,
University of Delaware, Newark, DE 19716.
USA.

rajagopa@cis.udel.edu

Chien-Chung Shen

Department of Computer and Info. Sciences,
University of Delaware, Newark, DE 19716.
USA.

cshen@cis.udel.edu

ABSTRACT

Past research studying the operations of TCP over wireless/mobile ad hoc networks has shown that TCP cannot be adopted as-is for use in mobile ad hoc networks (MANET) to achieve reasonable performance. This is because the assumptions made during designing TCP were made to make TCP work for wired networks, not MANETs. Consequently, most past research evaluating the performance of MANET routing protocols has concentrated on using CBR traffic over UDP for evaluation. In this paper, we distinguish practical use of TCP from use of TCP for evaluation purposes, and examine whether using *TCP as an evaluation tool* to evaluate MANET routing protocols will yield any new insights, even though we readily agree with past research that using TCP as-is (that is, as packaged for wired networks) for practical use over MANET is not a good approach. We motivated this idea because even though TCP does not function well over MANET, the fundamental expectations TCP has over the network are “universal” metrics which MANET researchers should strive for as well.

Using simulations, we show that by studying the performance of MANET routing protocols using TCP traffic, we obtain a *different* understanding of the protocols studied, and this understanding cannot be obtained via applying CBR/UDP traffic. In this spirit, we urge MANET routing protocol designers *to use TCP as an evaluation tool* during the protocol design process.

Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design—*Wireless communications*

General Terms

Algorithms, Performance, Design

*This work is supported in part by National Science Foundation under grants CNS-0335302 and CNS-0347460.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

IWCMC'06, July 3–6, 2006, Vancouver, British Columbia, Canada.
Copyright 2006 ACM 1-59593-306-9/06/0007 ...\$5.00.

Keywords

TCP, routing protocol evaluation, metrics, ANSI, AODV, fairness index, congestion windows, mobile ad hoc networks

1. INTRODUCTION

TCP performance over wireless and mobile ad hoc networks has been studied extensively in the past, for example, [3, 17, 18]. Some of the past transport layer research for MANET have shown that TCP is a bad protocol to use over MANET, for example, [6, 4, 5] and some transport layer research in MANET has also focused on improving TCP for MANET, for example, [16, 1, 2].

Regardless of the above, what past research shows is that TCP cannot be used for practical purposes over MANET as-is. This is because the fundamental operations of TCP functions (such as congestion/flow control and fairness and reliability improvement mechanisms) were designed for wired networks, rather than MANET. For instance, packet drops result only from congestion in wired Internet whereas both congestion and link failures (due to mobility and interference) result in packet drops in MANETs. Most of the past research evaluating routing protocols for MANET, consequently, has concentrated on using CBR traffic over UDP for evaluation, for example, [11, 8, 9].

In this paper, we distinguish practical use from use for evaluation and we are concerned with whether or not using *TCP as an evaluation tool* for MANET routing protocols will yield new insights into the protocol design, even though *we readily agree with past research that using TCP as-is (that is, as designed for wired networks) for practical use over MANET is not a good approach*. We motivated this idea because even though TCP does not function well over ad hoc networks, the fundamental expectations TCP has over the network, such as the expectation of congestion, feedback mechanisms using a bi-directional path, fairness of flow (using flow-control at the source/sink), fairness of bandwidth distribution in the presence of multiple flows etc., are “universal” metrics which MANET researchers should strive for as well. Thus, by using TCP over MANET and understanding the performance of the routing protocols for MANET, we will gain a *different*, more higher-level, “per-flow” understanding of our protocols. Such an understanding of the routing protocols is not possible when we apply CBR/UDP loads for evaluation.

In addition to the above, we are also motivated by the fact that the routing layer is the first layer where a notion of destination exists (as compared to a notion of next hop).

Thus, the first layer to have explicit end-to-end purviews (TCP) will be affected by the routing protocol. Thus, we see that that deficiencies pertaining to the end-to-end-ness of the routing protocol can only be pointed out using TCP flows for evaluation and we should try to improve the routing protocol's behavior such that it affects TCP performance favorably.

Lastly, we are motivated by the fact that the performance of a UDP network varies *fundamentally* from the performance of a TCP network for the same settings. We see this performance difference in Figures 1(a)–1(d). These results are for two AODV [11]¹ networks in identical environments², and differing only in the application used. In both networks, the application layer sends x pkts/s from 25 data sources for 220 seconds, starting at 40 seconds into the simulation. We see how the UDP network consumes vastly more MAC resources (shown as the number of MAC layer unicasts and broadcasts) for the smaller number of packets received at the receiver's application layer, and also ends up receiving an increasingly lower percentage of packets sent as the application layer packet rate increases despite the increase in throughput. UDP is able to achieve better throughput because it does not perform flow/congestion control, but owing to the same reasons, it also chokes the network, making inefficient use of MAC resources, judging by the amount of MAC resources consumed per packet delivered to the application layer. Thus, as the packet rate increases, UDP's reckless use of resources in the network comes with heavy penalties. The TCP network, on the other hand, places loads in the network carefully—by gaging the amount of data actually received at the receivers, the TCP network regulates the rate at which packets are pumped into the network and buffers the rest. Owing to this, the TCP network is both less congested, and, by the same token, makes more efficient use of MAC resources. The reason why TCP delivers fewer packets than expected ($220 \times 25 \times x$) is because of buffer overflows. Clearly, we see that the UDP approach to evaluating routing protocols will force designers to assume the “worst case” during routing protocol design.

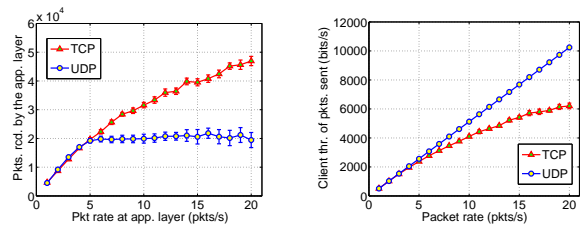
The rest of the paper is organized as follows: In the next section, Section 2, we discuss how the MANET environment affects TCP and how TCP affects the routing protocol in a MANET, followed by what metrics we can study when using TCP. In Section 3, we discuss our experimental results and insights when using TCP over MANET routing protocols. In Section 4, we conclude our paper with a note on future research.

2. TCP AND THE ROUTING PROTOCOL

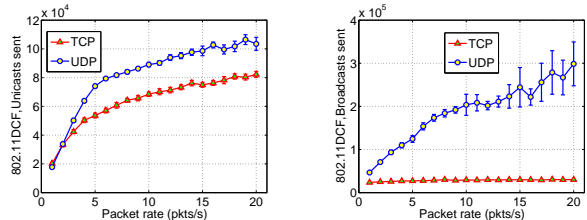
A discussion of the various aspects of TCP operations, such as TCP's flow-control, congestion-control/avoidance, fast retransmit/recovery operations can be found in [15].

¹Ad hoc On-demand Distance Vector (AODV) routing protocol for MANET is a popular routing protocol for MANET.

²The network is a 50-node network in a 1100m \times 1100m terrain using 802.11b at the physical layer (with a 250m transmission range) at 2Mbps and 802.11DCF at the MAC layer. The mobility model is Random Waypoint with a maximum speed of 20m/s and minimum speed of 0.001m/s and a pause time of 10s. Our AODV model used default parameters, and processes HELLO messages (sent every 1s), performed local repair, and retried route requests twice.



(a) Number of packets received at the application layer. (b) Throughput at the application layer.



(c) 802.11DCF, Unicasts sent. (d) 802.11DCF, Broadcasts sent.

Figure 1: AODV performance over TCP loads and UDP loads.

2.1 How does the MANET environment affect TCP?

In [10], the authors summarize the various issues of using TCP over MANET. Most of the issues summarized in [10] are related to routing protocol design and should be addressed at the routing layer if the MAC layer is unable to address them. The following list, adapted from [10], lists the various issues pertinent to routing protocols running under TCP:

1. The MANET environment is characterized by frequent path breaks.
2. The MANET environment is characterized by frequently changing path lengths of routes which affect the throughput capacity of the MANET [3].
3. Some or all links may suffer from asymmetric link behavior in a MANET.
4. Some routing protocols for MANET set up only a unidirectional path during the route discovery phase.
5. In some cases, multipath routing can be used for alleviating the effects due to hotspots in a MANET environment, but this has been shown to be not always effective [17].
6. Some MANET protocols can handle network partitioning and re-emerging, for example, [14], but these protocols may potentially need unbounded buffer spaces to function effectively.

All of the above issues directly affect the path setup and maintenance aspects of a MANET, thus affecting TCP performance.

2.2 How are routing protocols affected under TCP loads?

During the connection establishment process, TCP expects a bi-directional path to be set up and maintained (by the routing protocol) for the duration of the connection. This bi-directional path allows for the smooth flow of

data/ACKs and the maintenance of end-state information at the end points of the connection. Routing protocols that establish connections only in one direction thus incur additional overheads under TCP loads and increase the setup time for the TCP connection. Reactive routing protocols should set up the path in both directions (source \leftrightarrow sink). With proactive protocols, this is not an issue, but both classes of protocols should *maintain* the path and react *often enough* so that TCP, operating above the routing layer, does not lose data/ACK packets due to frequent congestion (due to the creation of hotspots) or frequent path breakages common in a MANET environment. If the routing protocol does not react quickly enough or does not maintain the state information, TCP performance drops.

For example, if a link breaks in the path between source \leftarrow sink during a connection, ACKs will be lost. TCP responds by reducing the congestion window (because it perceives congestion in the network) so fewer packets are sent into the network. Of course, if the routing protocol responds quicker than TCP in recognizing congestion or link breakage, then, TCP is oblivious to the problem, because the routing protocol has repaired the path. Otherwise, TCP’s response of congestion window reduction will reduce the throughput of the connection. Thus, a routing protocol not reacting quickly enough to topology change and congestion exposes TCP to the adverse effects of link breakage and congestion in the network.

2.3 Some metrics for studying TCP loads over MANET routing protocols

Using the insight gained by the issues discussed above we try to answer the following questions:

- Q.1:** What metrics of a TCP-driven network do we study to understand what aspect(s) of the routing protocol?
- Q.2:** Why can’t we study these metrics of the routing protocol by using UDP loads?
- Q.3:** Why are those metrics important?

We address the first question on a case-by-case basis. For **Q.3**, we note that the expectations which TCP has over the network are “universal” metrics which routing protocols for MANETs should also strive to provide for the MANET transport layer. Thus, if TCP unearths some problems in the routing protocol, these are problems which the routing protocol designer should try and address for effective routing. For **Q.2**, we note that by using TCP loads, we are able to study “per-flow,” end-to-end metrics in the context of reliable and fair data delivery, and thus these metrics cannot be studied under CBR/UDP loads.

We now discuss some metrics for studying TCP loads over routing protocols for MANETs.

2.3.1 Fairness index of end-to-end metrics

Packet delivery ratio (the ratio of the number of packets received to the number of packets sent) at the transport layer, a metric commonly used when studying a routing protocol under UDP/CBR flows, is a misleading metric when studying TCP flows, because the aim of TCP is to restrict the flow from the sender (source) such that the delivery rate at the TCP layer is high at the sink. To this effect, the TCP sender may drastically reduce the number of packets sent to the sink to maintain a high packet delivery ratio. Thus, this

metric by itself is not a good indicator of performance, and is only meaningful when both the number of packets sent and the packet delivery ratio are high.

Under TCP flows, measuring the end-to-end metrics such as number of packets received at the sinks, number of packets sent at the senders, end-to-end throughput and calculating the *fairness index* of these metrics help us understand how TCP reacts to the network conditions. This, in turn helps us understand how the routing protocol has assisted TCP operations.

The Jain fairness index [7] of a sample $X = \{x_1, x_2, \dots, x_n\}$, \mathcal{F}_X , is defined as follows:

$$\mathcal{F}_X = \frac{(\sum_{i=1}^n x_i)^2}{n \cdot \sum_{i=1}^n (x_i)^2}$$

such that $\mathcal{F}_X \in (0, 1)$. An \mathcal{F}_X close to 1 indicates that the measured metric has high fairness.

Some end-to-end metrics that can be measured for fairness are

1. The number of packets received at sinks, measured across all TCP data sinks,
2. The throughput at the data sinks across all data sinks, and
3. The connect times measured at the data sources, across all data sources indicating how the data sources vary in their response times.

The fairness index of the observed metrics across the data sinks/sources (ends of the TCP connection) should be as high as possible. High variation in the number of received packets, throughput etc. result in low fairness indices for these quantities and suggest that the TCP flows have not “fairly” divided the bandwidth among them, thus indicating the failure of the routing protocol to do load balancing. Cross-layer information, such as IP queue length, may be used by the routing protocol to efficiently divide the flows in the network and prevent the creation of hotspots.

2.3.2 TCP connect time

TCP connect times have been used before to evaluate the performance of routing protocols under (modified) TCP in a MANET, for example, in [3]. The authors in [3] employ a fixed RTO (retransmission timeout) scheme to help the MANET environment distinguish between node unreachability (due to temporary partitions) and congestion at the next hop. In using un-modified TCP as a tool for evaluating routing protocols for MANETs, low connect times indicate a routing protocols’ resilience to both congestion and node unreachability under which conditions, TCP does not spend too much time in connection establishment.

2.3.3 Measuring congestion window growth

Measuring the rate and extent of the congestion window growth and studying the congestion window-simulation time graph of the connection allows us to accurately study the congestion along the connection during the connection.

Measuring congestion windows for routing protocols has been done previously in literature, for example, in [3]. TCP starts a session using slow start (exponential `snd_cwnd`³ growth

³TCP controls the rate and extent of its congestion window growth by manipulating the *send congestion window*, `snd_cwnd` and the *slow start threshold*, `ssthresh`.

until `snd_cwnd > ssthresh`), but TCP reacts to the (perceived) packet loss by switching to congestion avoidance (by manipulating `ssthresh`, resulting in linear `snd_cwnd` growth). In Figure 2, we see how both ANSI [12, 13], a congestion-aware, hybrid, ant-based protocol, and AODV [11] have switched to congestion avoidance almost immediately after the start of the TCP session (40 seconds)⁴.

In Figure 2, we see that ANSI is able to steadily grow its `snd_cwnd` to the maximum (16384 bytes), while AODV suffers from repeated losses (as seen from the number of timeouts, which is more than 10). Note that ANSI does not suffer even one timeout during the entire simulation (TCP connection closes at 260 seconds). ANSI is able to maintain the a non-congested route from the data source to the data sink consistently and efficiently, owing to its congestion-aware behavior, thus steering IP from using next hops and paths which are congested. AODV, however, reacts too slowly to congestion and link breakage, thus exposing TCP to the issues due to link breakage and congestion.

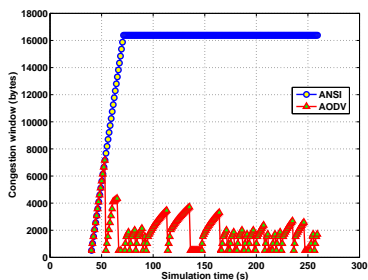


Figure 2: ANSI vs. AODV: congestion window growth for one TCP sender for the outgoing (headed towards the data sink) stream in a hybrid network containing both fixed infrastructure (with multiple interfaces, 802.11b and Ethernet) and MANET nodes.

If the congestion window growth under a routing protocol shows a predominantly slow starting TCP, then the network using the routing protocol is obviously not congested. On the other hand, congestion avoidance behavior can also indicate low congestion levels if the congestion window grows to the maximum. Frequent timeouts (resetting the congestion window to 1 MSS) indicate severe congestion and link breakage in the network, and point to the lack of resilience of the routing protocol.

3. STUDYING TCP FOR SOME COMMON ROUTING PROTOCOLS

In this section, we study the performance of two routing protocols for MANET, AODV [11] and ANSI [12, 13] using TCP flows and provide some insights into the working of these protocols. We used QualNet version 3.7 for the simulations. AODV is a common reactive routing protocol [11], and ANSI is an ant-based, congestion-aware routing protocol. We chose to compare AODV and ANSI because we wanted to see how these two protocols, which perform well under UDP loads⁵, perform under TCP loads. In addition,

⁴Details of ANSI can be found in [12, 13], and have been omitted here for space considerations.

⁵For a comparison study of ANSI vs. AODV under UDP

we wanted to test if congestion-awareness of ANSI is a desirable property for routing protocols in MANET.

ANSI protocol models, physical, MAC layer characteristics are identical to the ones used in [12]. For AODV, neighbor discovery mechanisms using HELLO beacons was enabled. The beacon timer was 1s. We enabled local repair for AODV. The maximum number of RREQ retries for AODV was set at 2. For both protocols, nodes use a 100MB buffer for protocol operations.

We used omnidirectional 802.11b antennas at the physical layer with a transmission range of 250m and 802.11DCF at the MAC layer. Also, the physical layer used a two-ray pathloss model and no propagation fading model. The link bandwidth was 2Mbps. We used the random waypoint model with non-zero minimum speed (to prevent speed decay) and 10s pause time for mobility.

We performed three experiments in which we used 25 concurrent TCP streams under Super application in Qualnet, which is an application used to generate constant rate of application layer traffic between a source and a sink. We used TCP-NewReno with a maximum send/receive buffers of 16384 bytes. In all the experiments, 50 nodes with one 802.11b interface were used.

In Experiment 1, we varied the packet rate of the application from 1 packet/s to 20 packets/s in steps of 1 packet/s. In Experiment 2, we varied the speed of the nodes in the network from no mobility (static) to 20 m/s in steps of 4 m/s. To alleviate the problems due to speed decay and understand the effects due to mobility, we set both the minimum and maximum speed in the mobility model to be the same so we did not suffer from the speed decay problem. In Experiment 3, we varied the size of the network from a square network to an increasingly rectangular network to study the effects due to increasing hop count in the path from the source to destination.

In Experiment 1 and 3, the minimum waypoint speed was 0.001 m/s and the maximum waypoint speed was 20 m/s. Also, for Experiments 2 and 3 we used a packet rate of 10 packets/s. In Experiment 1 and 2, the size of the terrain was 1100m×1100m. In Experiment 3, we varied the terrain size such that one side of the terrain was an integer fraction of 1100m and the other side such that the total area was 1100m×1100m.

We performed 20 trials of all our experiments (to account for stochastic aberrations). We report our findings below with graphs showing 95% confidence intervals of all the observed metrics.

3.1 Results

Figures 3(a) – 3(d) show the results of Experiment 1; Figures 4(a) – 4(d) show the results of Experiment 2; and Figures 5(a) – 5(d) show the results of Experiment 3. We did not show the throughput measurements and the fairness of throughput across receivers because the trends of throughput and fairness of throughput are similar to the trends in packet delivery and fairness of packet delivery.

3.1.1 Packets received and fairness of packets received

From the figures for experiments 1, 2 and 3, we see that in general, ANSI is able to deliver more packets to the receiver in the presence of multiple TCP streams. In addition, ANSI loads, please see [12, 13].

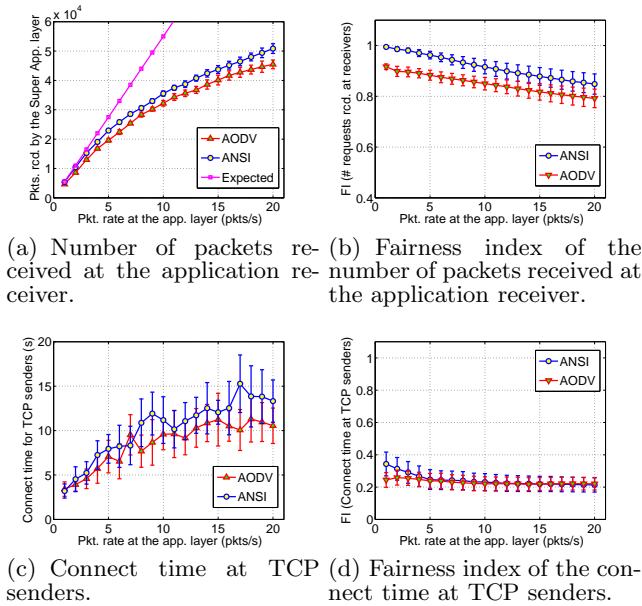


Figure 3: Performance of ANSI and AODV with increasing packet rate.

has a higher fairness index of the number of packets received (in the case of Experiment 3, comparable⁶ fairness), across the data sinks than AODV.

As the packet rate increases, the congestion in the network increases, thus both protocols show a decreasing percentage of the packets being delivered. As the possibility of collisions increase, despite the routing protocol’s best efforts, it is increasingly difficult to distribute network resources evenly amongst flows. This is why the fairness index decreases as the packet rate increases.

When the speed increases, we see that the number of packets received decreases slightly for ANSI and increases slightly for AODV. Increasing rate of link breakage (with increasing speed) has a different effect on either protocol. For ANSI, this results in more aggressive congestion-aware routing, thus increasing hop length but decreasing packet delivery. For AODV, this translates to more aggressive route discovery activity, improving the chances of packet delivery. ANSI shows an improvement in fairness as speed increases owing to more aggressive congestion aware routing (which helps in load balancing), and AODV shows an improvement in fairness because of more aggressive route discovery activity.

As the network becomes squarer, both ANSI and AODV are able to deliver more packets owing to shortening in the number of hops to the data sink. This also makes more nodes available between a source and sink to perform routing activities. For the same reason, the contention for MAC layer resources decrease as the network becomes squarer, thus improving the fairness index.

Despite the fact that ANSI is fairer and more successful in delivering more packets as compared to AODV, both pro-

⁶Two protocols A and B are “comparable” when the confidence intervals of their average values overlap [7]. In the same vein, protocol A is “better” than protocol B when the average value of A is better and the confidence intervals of A and B do not overlap.

ocols suffer *significant* losses as the packet rate increases. A total of $220 \times 25 \times x$ packets (where x is the packet rate) is expected to be delivered by the Super application, but the fraction of the data actually delivered decreases at an increasing rate in both ANSI and AODV (see Figure 3(a) – the straight line (with square markers) shows the number of packets to be delivered ideally). We see that speed does not affect the number of packets received significantly (see Figure 4(a), 55000 packets expected) but terrain size affects it significantly (see Figure 5(a), 55000 packets expected) – more rectangular terrains decrease the number owing to increasing number of errors along a longer path.

ANSI is also able to send more packets (at TCP senders) as compared to AODV for all three experiments, resulting in a higher packet delivery metric (the graphs for the number of packets sent is not shown owing to space constraints) for ANSI.

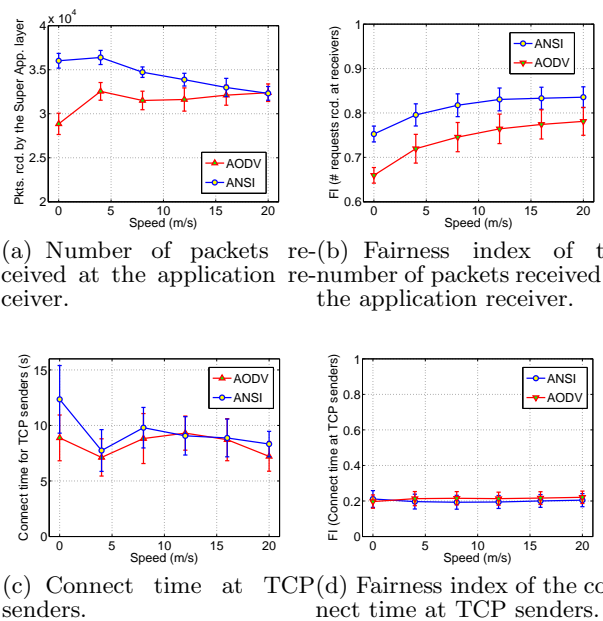
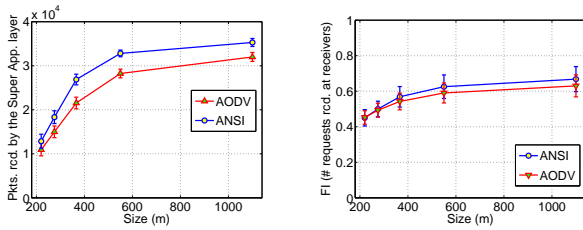


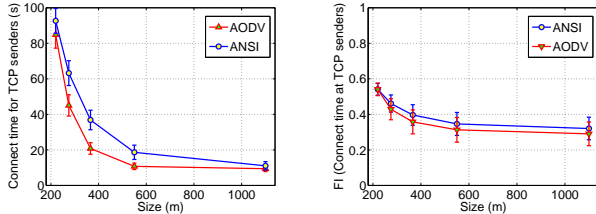
Figure 4: Performance of ANSI and AODV with increasing speed.

3.1.2 Connect times and fairness of connect times

For experiments 1 and 2, ANSI and AODV are comparable in terms of connect times and fairness of connect times. For Experiment 3, ANSI shows higher connection times. For both ANSI and AODV, connect times increase as the packet rate increases and when the network becomes more rectangular, but fluctuates as the speed increases. This is because of increased congestion in networks when the packet rate increases. When the speed increases, connect times reflect how aggressively either protocol performs route discovery activity. For ANSI, more next hop choices are available for each destination, decreasing connect times. When the network is very rectangular, both the ANSI network and AODV network show large connection times, owing the high path lengths. Rectangular networks also show a higher fairness in connection times as compared to squarer networks. This is because in a squarer network, paths tend to cross each other more than in a rectangular network, which increases



(a) Number of packets received at the application receiver. (b) Fairness index of the number of packets received at the application receiver.



(c) Connect time at TCP senders. (d) Fairness index of the connect time at TCP senders.

Figure 5: Performance of ANSI and AODV with increasing squareness of terrain.

the contention. Regardless, both ANSI and AODV show *very low* fairness for connect times.

3.2 Discussion

More packets are sent from the data source at a higher rate and more packets are in turn received at the data sink in ANSI. Consequently, we see that the congestion awareness of ANSI does improve fairness of the number of packets received metric in an ANSI network. Thus, we see that there is a strong correlation between performing congestion-aware routing and improved packets received metrics. The fact that ANSI is fairer with respect to packets received metrics indicates that the ANSI network is able to divide the network resources more evenly between flows.

Lastly, even though we see that ANSI is able to perform better and fairer than AODV owing to congestion-awareness, we see that both protocols are not upto the mark when it comes to connect time metrics—a measure of how responsive the protocols are. Both protocols show wide fluctuations between receivers in the network. Besides, when the network is very rectangular, the fairness of connect time seems to be higher than the fairness of packet delivery (though the connect times are very high), and the condition reverses with increasing squareness of the network. This is a difficult condition for protocol designers to work around.

4. CONCLUSIONS

We have shown that using TCP to evaluate routing protocols for MANET has the benefit of illustrating the effect of the routing process in an end-to-end, rather than hop-to-hop context, thus yielding new insights into the working of routing protocols for MANETs. We note that using UDP would not have yielded this “per-flow” understanding and how evenly the flows divide the resources. Thus, we see that even though TCP is not a useful protocol *as-is* for practical use over a MANET, when used as an evaluation tool, it gives

us a new way to design and understand routing protocols for MANETs.

In particular, we see that performing congestion-aware routing can improve some metrics in a MANET, but it remains to be seen about how we can incorporate our insights into the routing protocol design for improving MANET routing protocol metrics. Also, it can be seen that the effect of TCP over other lower layers in the MANET will also provide new insights regarding the design of these protocols. In this spirit, we call on the MANET research community to rethink the use of TCP for evaluating their protocols for MANET.

5. REFERENCES

- [1] K. Chen, Y. Xue, and K. Nahrstedt. On setting TCP’s congestion window limit in mobile ad hoc networks. In *ICC*, May 2003.
- [2] R. de Oliveira and T. Braun. A Dynamic Adaptive Acknowledgment Strategy for TCP over Multihop Wireless Networks. In *INFOCOM*, March 2005.
- [3] T. D. Dyer and R. V. Boppana. A Comparison of TCP Performance over Three Routing Protocols for Mobile Ad hoc Networks. In *ACM Symposium on Mobile Ad hoc Networking and Computing, MobiHoc*, October 2001.
- [4] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla. The Impact of Multihop Wireless Channel on TCP Throughput and Loss. In *INFOCOM*, April 2003.
- [5] M. Gerla, R. Bagrodia, L. Zhang, K. Tang, and L. Wang. TCP over Wireless Multihop Protocols: Simulation and Experiments. In *ICC*, 1999.
- [6] G. Holland and N. Vaidya. Effects of Multipath Routing on TCP Performance in Ad Hoc Networks. In *Globecom*, November 2004.
- [7] R. Jain. *The art of computer systems performance analysis: Techniques for Experimental Design, Measurement, Simulation and Modeling*. John Wiley and Sons, 1991.
- [8] Y.-B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. In *Mobile Computing and Networking*, pages 66–75, 1998.
- [9] S. J. Lee, W. Su, and M. Gerla. On-demand multicast routing protocol in multihop wireless mobile networks. *Mob. Netw. Appl.*, 7(6):441–453, 2002.
- [10] C. S. R. Murthy and B. S. Manoj. *Ad hoc Wireless Networks: Architectures and Protocols*. Prentice Hall Communications Engineering and Emerging Technology Series, 1999.
- [11] C. E. Perkins, E. M. Royer, and S. R. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. Internet Draft (draft-ietf-manet-aodv-09.txt), November 2001. Work in Progress.
- [12] S. Rajagopalan and C.-C. Shen. ANSI: A Unicast Routing Protocol for Mobile Ad hoc Networks Using Swarm Intelligence. In *International Conference on Artificial Intelligence (ICAI)*, Las Vegas, NV, USA, June 27–30 2005.
- [13] S. Rajagopalan and C.-C. Shen. ANSI: A Swarm Intelligence-based Unicast Routing Protocol for Hybrid Ad hoc Networks. *Journal of Systems Architecture*, Special Issue on Nature Inspired Applied Systems, 2006. To appear.
- [14] C.-C. Shen, S. Rajagopalan, G. Borkar, and C. Jaikao. A flexible routing architecture for ad hoc space networks. *Computer Networks*, 46(3):389–410, October 2004.
- [15] W. R. Stevens. *TCP/IP Illustrated, Volume 1 – The Protocols*. Addison Wesley Professional Computing Series, Reading, Massachusetts, 1994.
- [16] F. Wang and Y. Zhang. Improving TCP Performance over Mobile Ad-Hoc Networks with Out-of-Order Detection and Response. In *The ACM Symposium on Mobile Ad hoc Networking and Computing (MOBIHOC 2002)*, Lausanne, Switzerland, June 9–11 2002.
- [17] Z. Ye, S. V. Krishnamurthy, and S. K. Tripathi. Effects of Multipath Routing on TCP Performance in Ad Hoc Networks. In *Globecom*, November 2004.
- [18] Z. Ye, S. V. Krishnamurthy, and S. K. Tripathi. Use of Congestion-Aware Routing to Spatially Separate TCP Connections in Wireless Ad hoc Networks. In *The 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS*, October 2004.