# TCP Performance with Active Dynamic Source Routing for Ad Hoc Networks

*Yu He, Cauligi S. Raghavendra*
Department of Computer Science
University of Southern California
{yuhe, raghu}@usc.edu

*Steven Berson, Bob Braden*
Information Sciences Institute
University of Southern California
{berson, braden}@isi.edu

## Extended Abstract

A wireless ad hoc network consists of mobile nodes that communicate among themselves using radios, without the aid of a fixed networking infrastructure. An important research problem is realizing reliable end-to-end data transmissions, like the one provided by TCP in the Internet. It is known that TCP does not work well in ad hoc networks ([1], [2], [3], and [4]) because of high bit error rate (BER) in wireless channels, and high frequency of route failures and changes. High BERs and route failures result in continuous loss of TCP data segments or acknowledgements. In addition, route failures cause routes to be re-discovered. If the route re-discovery takes longer than the TCP retransmission timeout (RTO), retransmission is triggered and the RTO backs off exponentially. The consequence is severe reduction in throughput even when there is no congestion. Furthermore, if the new route is significantly different from the previous route (e.g., with different route length), it is possible to have out-of-order packets arriving at the receiver, which could mistakenly trigger congestion control, further negating TCP performance.

In this paper, we propose an active networking approach to improve TCP performance. Our approach is based on reducing route failures in Dynamic Source Routing (DSR [5]). The approach, called Active DSR (ADSR), updates stale routes more quickly than DSR. Thus route failures are reduced, route changes take less time, and TCP RTO backoffs are significantly reduced. Furthermore, ADSR supports congestion detection by monitoring queues in each node. If the queue at one node has a length larger than some threshold, ADSR computes an alternate route for those flows that pass such node. Thus congestion is predicted and avoided at potentially congested nodes.

ADSR makes use of active networking techniques [6] by providing an active helper to support the traditional DSR. This helper is loaded when an active packet arrives at or originates from a node and then operates on the active packet. The active packet is periodically generated by a randomly-chosen node. The basic idea is to have the active helper drive the active packet to visit each node twice periodically. The first visit is to obtain topology information of the network; the second visit is to update route caches using the obtained information. The payload of the active packet is a connection matrix for the network topology. During the first visit of the active packet, the active helper checks the nodes' neighbor information and updates the connection matrix of the packet. Thus after the first visit, the active packet contains the topology information of all nodes it has traveled. Then during the second visit, the active helper at each node validates and updates its route cache according to the connection matrix in the packet. In the validation phase, each routing entry is checked against the connection matrix and those that disagree are removed. In the updating phase, the active helper adds routes learned from the active packet to route caches. In this way, route cache miss rates are decreased and route flooding triggered by both stale routes and new routes are reduced [7].

Furthermore, ADSR positively respond to potential congestion by having the active helper also obtain the queue-length information during the first visit of the active packet. The value is evaluated by the active helper and a value indicating if the node is congested or not is added into the connection matrix. Thus after the first visit, the active packet contains a view about the traffic condition of all nodes it just went through. Then during the second visit, the validating and updating phases are extended as follows: in the validating phase, in addition to stale routes, those that contain nodes marked as potentially congested are also removed; in the updating phase, if a route to be added contains such a potentially congested node, an alternate route is computed and added. ADSR always looks for an alternate route with the same length. Since the new route has the same length as the old one, out-of-order packets caused by such route changes are reduced. In this way, the potentially congested nodes are detected after the first visit by active packets, and those bottleneck nodes are circumvented after the second visit. Thus congestion is effectively reduced with ADSR.

There are two parameters in our approach that affect the performance of ADSR. The first is the interval value for the initial timer, which is used to initiate the active packet visits. Each time the initial timer expires, an ADSR packet is generated. There is a tradeoff in choosing the interval of the initial timer - the shorter the interval, the more frequent the updates, and the more recent the route cache entries. Thus the route flooding should be less frequent and TCP traffic should be less likely to be interrupted by route failures. However, a shorter interval also means that there are more active packets roaming around the network. The value of the initial timer should be chosen to keep data rate high and routing overhead low. The second parameter is the threshold value for queue sizes. Obviously it should be lower than the queue length limit. However it should also approximate the value of the limit since otherwise there would be too much unnecessary route changes that would degrade the system performance. But if it were too close to the limit value, some potential congestion could be ignored (for example, congestion by bursty traffic could be neglected).

To improve the visit efficiency, the active packet can be restricted to visit each node only once, instead of twice. The active helper then updates route caches along the way of the visit by using partial topology information in the connection matrix. In a large ad hoc network, the active helper can set a TTL field in the active packet to limit how far it can go. Further, the active helper can be location aware and takes advantage of location-aided routing techniques [8]. For example, an ad hoc network area can be divided into subareas. In each subarea, one node is chosen to load the active helper. The helper then computes an appropriate routing curve [9] (for example, a circle) so that most nodes in the subarea can be visited by the active packet. Location-aided ADSR is one future research topic.

We have implemented ADSR in ns-2 [10] with CMU's wireless extensions [11]. To evaluate the performance of ADSR, we test TCP throughput and routing packet overhead. TCP *throughput* counts all TCP data segments successfully transmitted during a simulation session. *Routing packet overhead* is reflected by the routing packet number that is the rate of all DSR and ADSR packets injected by nodes during a simulation session.

Our simulation runs can be roughly classified into two parts. The first part is to test TCP performance under ADSR when there is no congestion in the networks. The second part is to see how ADSR responds to congestion in the networks. In our simulations, various network and traffic scenarios are considered. Extensive simulation results show that ADSR significantly improves TCP performance under various conditions (up to **72%** when no congestion in networks, and up to additional **25%** when there is congestion). Besides, since ADSR reduces route discovery flooding, these TCP improvements involve less routing packet overhead than the traditional DSR.

# REFERENCES


[1] *Jian Liu and Suresh Singh*, "Atcp: Tcp for mobile ad hoc networks," *IEEE J-SAC*, 2001.
[2] *G. Holland and N. Vaidya*, "Analysis of tcp perfomance over mobile ad hoc networks," *Mobicom*, 1999.
[3] *K. Chandran, S. Raghunathan, and S. Venkatesan*, "A feedback based scheme for improving tcp performance in ad-hoc wireless networks," *Proceedings18th Intl. Conf. on Distributed Computing Systems (IDCS'98)*, May 1998.
[4] *Imran Ali and et al*., "Energy efficiency and throughput for tcp traffic in multi-hop wireless networks," *Infocom*, June 2002.
[5*] D. B. Johnson and et al*., "The dynamic source routing protocol for mobile ad-hoc networks," *IETF Internat Draft http://www.ietf.org/internet-drafts/draftietf-manet-dsr-05.txt*, 2001.
[6] *D. L. Tenenhouse, J. M. Smith, W. D. Sincoskie, D. J. Wetherall, and G. Minden*, "A survey of active network research," *IEEE Communication Magazine*, January 1997.
[7] *Yu He, C. S. Raghavendra, Steve Berson, and Bob Braden*, "Active packets improve dynamic source routing for ad-hoc networks," *IEEE OPENARCH short paper session*, June 2002.
[8] Y. *Ko and N. H. Vaidya*, "Location-aided routing (lar) in mobile ad hoc networks.," *Mobicom*, October 1998.
[9] *Badri Nath and Dragos Niculescu*, "Routing on a curve," *Hotnets I*, October 2002.
[10] *USC/Information Sciences Institute*, "Network simulator," *http://www.isi.edu/nsnam/ns/*.
[11] *CMU Monarch Group*, "Cmu monarch extensions to ns-2 simulatr," *http://monarch.cs.cmu.edu/cmu-ns.html*, 1998.