Estimating the Ad Hoc Horizon for TCP over IEEE 802.11 Networks

Christian Tschudin and Evgeny Osipov

Technical Report CS-2004-001
University of Basel
March 14, 2004

Abstract

In this report we explore the hypothesis of an "ad hoc horizon" for wireless multihop networks. This horizon is defined in terms of number of nodes and number of hops beyond which TCP based network services are not useful anymore. We show that with IEEE 802.11 this horizon is as narrow as 2 to 3 hops and approximately 15 nodes. These results are obtained by choosing optimal settings (e.g., link layer feedback but no mobility) and focusing on worst case figures. We discuss the reason for this horizon to form and draw conclusions how this horizon could be extended.

Keywords:
Ad Hoc networks, ad hoc routing, TCP performance, IEEE 802.11.

UNI
BASEL

Computer Science Department, Computer Networks Group, Bernoullistrasse 16, CH – 4056 Basel
Switzerland. Phone: +41 61 267 05 55 Fax: +41 61 267 05 59 URL: http://cn.cs.unibas.ch/

1 Introduction

Simulations are probably the most important investigation tool for ad hoc routing protocols in wireless networks. They have been used to demonstrate the scalability of ad hoc routing over IEEE 802.11 technology, exploring multihop networks with several hundred nodes and paths of ten and much more hops. However, developers working with real 802.11 equipment know that the corresponding findings are far too optimistic and bear little resemblance to reality.

Experiences with the Ad hoc Protocol Evaluation testbed APE [7] indicate that paths longer than 3 hops are not reliable enough for even modest network usage. By using simulation techniques but adopting a different interpretation strategy, we were able to produce the same pessimistic results and thus confirm the assessment from practical experiments. We call this limit the "ad hoc horizon" and define it as the region spanned by either the number of nodes or the number of hops beyond which TCP performance is not acceptable for ordinary end user tasks like web browsing. We estimate that this ad hoc horizon is currently located at 2 to 3 hops or 15 nodes for the combination of TCP and IEEE 802.11. The ad hoc routing protocol does not seem to be of importance for the formation of is horizon.

The difference between our interpretation of simulation results and other studies lies in the focus on TCP instead of constant bit rate (CBR) traffic. Moreover, we examine worst case figures instead of averaged results which can draw a favorable picture "in general" but which possibly hide severe "particular" problems. In order to do a conservative estimation of the ad hoc horizon, we make optimistic assumptions with regard to the settings and only then attempt to extract worst case behaviors.

This report is organized as follows. We first explain our methodology of designing and analyzing simulation experiments. In section 3 we report on the found measurements. We analyze and discuss them further in section 4 before concluding with section 5.

2 Methodology

This section describes our approach for isolating the region wherein a wireless ad hoc network delivers "useful" services. In a nutshell our methodology can be characterized as making optimal assumptions on the system that we want to test while focusing on the worst performances than can be produced in such settings. For example, we will introduce a family of static topologies and thus remove any bad effects due to mobility. On the other hand, we will not look at average performance figures but always select the poorest case that an unlucky end user might encounter. However, before we look at these two elements, we briefly review classic and recent simulation approaches.

2.1 Related Simulations Studies in Ad hoc Networks

The performance comparison of Broch et al. [3] has influenced many subsequent simulation studies in defining the target values and parameter choices for the ad hoc network research community. Among the three metrics examined in their study two of them relate to routing aspects (efficiency and optimality). As we will operate in a static environment, we do not further discuss these. The other metric of interest to us is the packet delivery ratio. A prominent choice is the use of a constant bit rate (CBR) data source which pumps data into the network regardless of the network's condition. The justification for this choice is not fully stringent[1]. Although the authors did some averaging (communication pattern and topologies) and randomization (waypoint model and jitter), they did not want to do so for TCP. In light of many findings on TCP's bad performance in ad hoc networks, it is fair to say their comparison study could not have been conducted with TCP, as it would have produced erratic results.

While Broch et al. considered networks of 50 nodes with actual paths from 3 to 8 hops, much longer routes have been investigated in the recent past. In [6], simulation results are reported for up to 10'000 nodes with routes of more than 100 hops. The nodes moved according to the same random waypoint model as above and CBR sources were used in this case too. The study, however, does not motivate this choice and mentions TCP not at all.

Both studies referenced here examine settings which are in sharp contrast with our finding of a very narrow ad hoc horizon. Although both studies mention conference, emergency or military scenarios as potential applications, they do not argue in which way these scenarios can be served by CBR traffic only.

What is unfortunate from an end user point of view is that although the TCP protocol will presumably be the most important transport protocol for his network use (mail, web, instant messaging etc), ad hoc routing protocols are not systematically examined for their fitness for TCP. On the other hand, many studies have been conducted that explore the dynamics of TCP in a wireless multihop environment. TCP performance degradation has been reported for example in [4, 5] and TCP's unfair behavior in [8]. The goal of these studies is mostly on the modification of the MAC or the TCP protocol in order to improve TCP's performance. In this report we put the focus on the consequences of TCP's problems for the study of ad hoc routing protocols itself. More specifically, we ask about the useful operation range for ad hoc routing protocols that might be imposed by the quirks of TCP.

2.2 The Family of Beam Star Scenarios

Simulation studies for ad hoc routing protocols, especially when it comes to comparisons and rankings, avoid choosing "special" scenarios and communication patterns. This helps to remove biased settings and avoids the tuning of a protocol for particular circumstances. In our case, however, choosing a special scenario has the status of a counter-proof as we are looking for worst case figures: If we can show bad performance in one scenario, it is possible that other scenarios exist that offer even worse conditions, which would only reinforce the findings.

In order to test our hypothesis of an "ad hoc horizon" we created a family of topologies called "beam star". A central node (which could be a gateway into the fixed Internet) serves as end or start point for all TCP sessions. From this central node, a varying number of beams of identical length emanate in a regular fashion. Figure 1 shows four instances of this topology family.

The communication pattern consists of parallel FTP session. Each beam is used for an FTP transfer where the central node establishes sessions with each beam head. This results in a moderate number of sessions (one per beam) when compared to the total number of nodes, especially for long beam lengths. The total number of nodes is given by the formula $1 + beams * beamlength$.

[1] Quote from [3] "As the goal of our simulation was to compare the performance of each routing protocol, we chose our traffic sources to be constant bit rate sources. ... We did not use TCP sources because TCP offers a conforming load to the network, meaning that it changes the times at which it sends packets based on its perception of the network's ability to carry packets. As a result, both the time at which each data packet is originated by its sender and the position of the node when sending the packet would differ between the protocols, preventing a direct comparison between them."

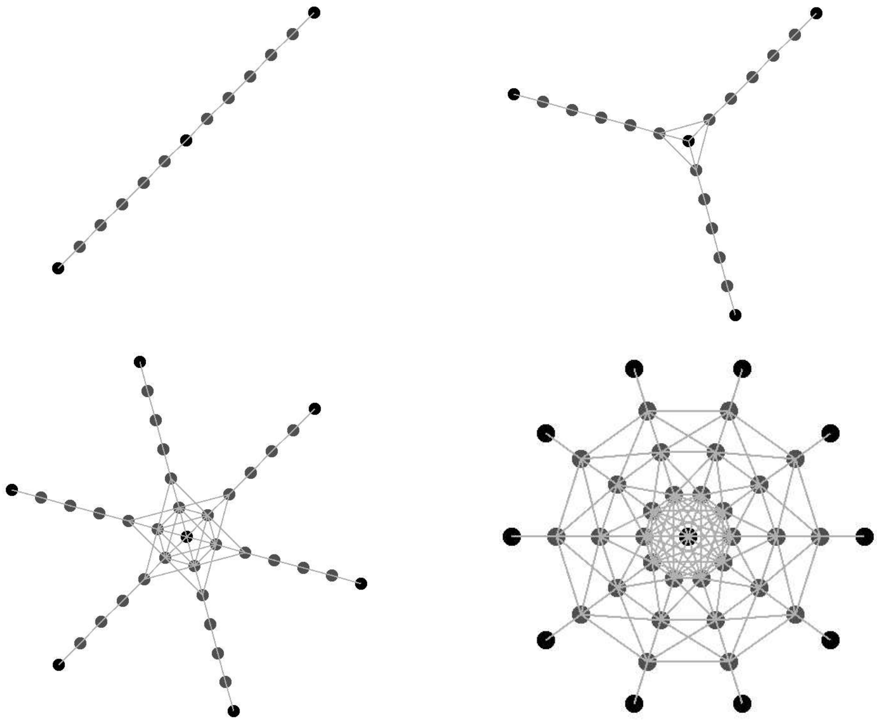

Figure 1: “Beam star” network topologies for various numbers of beams and beam lengths (2x6, 3x6, 6x6, 10x4) shown with potential connectivity. (The last figure is shown with a different scale: the internode distance along a beam is identical for all topologies.)

The family of beam star topologies permits us study the effect of increasing path lengths and growing number of nodes in a controlled manner. We expect both to have a negative impact on TCP performance. Thus, we can assess the combined limit for effective TCP data delivery in multihop ad hoc networks.

2.3 Worst Case Figures under Optimal Configurations

A “usefullness” metric from an end user’s point of view is the time during which a TCP session does not make any progress i.e., what the user perceives as a “frozen” connection. For each session we examine the sequence of TCP segments and record the time intervals in which TCP does not make progress i.e., during which the TCP protocol waits for the reception of a new segment that can be delivered to the application. In order to account for short bottlenecks in the network and to mimick some amount of user tolerance, we only count no-progress intervals longer than 3 seconds. These larger than 3 seconds intervals are accumulated and put into relation to the total duration of the test run. The higher this ratio is, the more sputtering was the session: A 100% *no-progress ratio* means a complete stall. For a given topology we compare the no-progress ratios of all TCP sessions and testruns and retain the highest ratio.

The other metric we examine is the *unfairness* among TCP session. We use the classic maxmin fairness index

$$f = \frac{(\sum_{i=1}^{n} x_i)^2}{n \sum_{i=1}^{n} x_i^2}$$

where x_i is the goodput in TCP segments of FTP session i. We reverse this value such that we get an unfairness index. A value of 0 means that all TCP sessions receive the same (although perhaps low) share of the network's total capacity, while an unfairness of 1 means that one session is monopolizing all bandwidth.

2.4 Measurement Setup

In our simulation with ns-2 we used the following settings:

IEEE 802.11 bandwidth transmission range interference range	2 Mbps 250 m 550 m
Routing protocol	AODV-UU 0.8 [1] (AODV v13), local repair enabled
Application	FTP, starting at t=5 sec
Simulation run number of repetitions	300 sec 6

The beamstar scenarios use an internode distance of 130 meters. Some resulting potential connectivity patterns can be seen for the topologies in Figure 1.

3 Results

Using the approach described above, we present a series of simulation results that first identify the ad hoc horizon and then explore to which extent this finding depends on choices made.

3.1 TCP No-Progress Time

As a base case for demonstrating the ad hoc horizon we chose a setting that is optimal in terms of an ad hoc network: Nodes are stationary, we use ns-2's simplified 802.11 model with a single 2 Mbps transmission speed for broadcast as well as unicast messages, AODV is permitted to use link layer feedback and all TCP sessions run over the same number of hops.

In this setting we extract the worst no-progress ratio among all TCP sessions as defined in Section 2.3. This ratio is then plotted against the number of beams and beam length, yielding a surface (see Figure 2). As a reading example, we note that with 2 beams an 3 hops (resulting in a network of 7 nodes) we have a ratio of 30%, hence more than 30% of the time the session appeared to be "frozen". Not visible from the graph, but by inspection of the traces, this no-progress time contains stalls of up to 17 seconds.

The main observation that we point out is that there is a marked slope when moving from 2 to 3 hops. In the other dimension – the number of beams – we get similarily bad figures for two hops when we reach the number of six or seven beams. Together, this corresponds roughly to a region

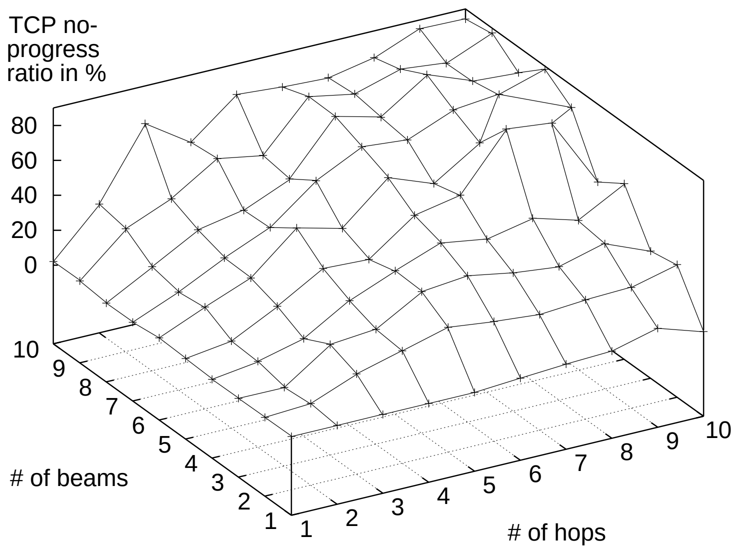

Figure 2: Worst accumulated TCP no-progress time for AODV, in dependency of the number of beams and their length.

of 15 nodes or 3 hops, whatever dimension is explored first, beyond which the TCP no-progress ratio is 30% or more. We call the border of this region the "ad hoc horizon". It is a horizon in the sense that we don't see useful TCP services beyond this range. For an end user this means that at the edge of the ad hoc horizon he has to anticipate that his TCP session might be stalled for one third of the time.

3.2 Disabeling Link Layer Feedback

The use of link layer feedback is popular in simulation but is not available in reality. To get an assessment of the impact of this choice (and to explain the frustration of practioners in the field), we ran the same simulation but disabeled link layer feedback for AODV.

Figure 3 (right subfigure) demonstrates the expected degradation in a drastic way: After 2 hops we litterally hit a wall. Instead of a 30% stall ratio this number jumps to 60%. Note that the left figure (for a range of 1–4 beams and 1–4 hops) is identical to the corresponding region in figure 2: In the case where AODV cannot use link layer feedback, we hit the 60% no-progress ratio already with 3 hops while with link layer feedback this plateau is reached only at about 6 beams and 6 hops.

3.3 TCP Unfairness

Next we examined "TCP fairness" as another metric for capturing the potential dissatisfaction an end user might experience. In the ideal case, all TCP sessions are treated fairly. Users would probably accept reduced TCP throughput and even some stalls as long as they are spread over all users in a equal way.

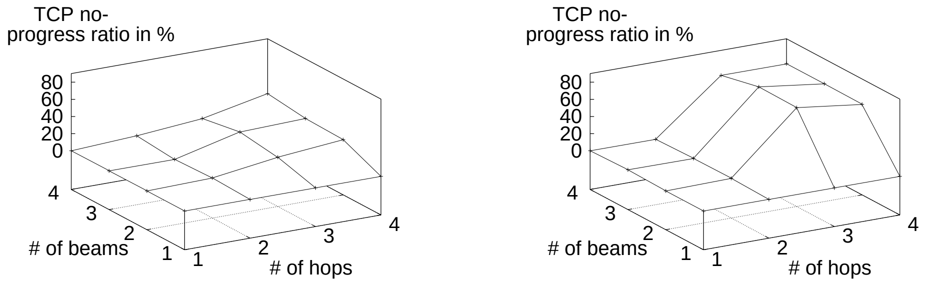

Figure 3: Comparing TCP no-progress time for AODV, with and without layer feedback.

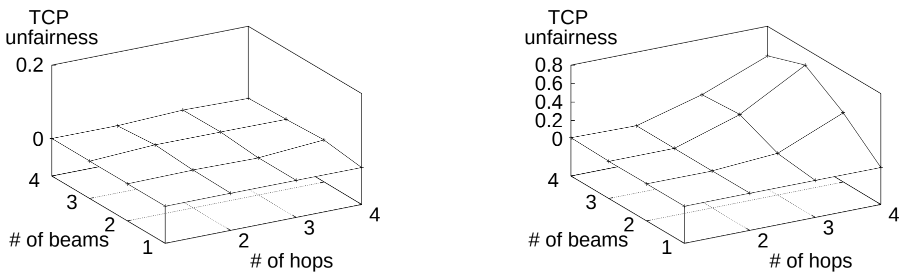

Figure 4: TCP unfairness for AODV, with and without link layer feedback.

In Figure 4 we plot TCP unfairness (with and without link layer feedback) i.e., how unequally the various TCP sessions are treated by the ad hoc network (see Section 2.3). As can be observed in the right subfigure where link layer feedback is disabled, we have again a slope starting at 3 or 2 hops, depending on the number of beams. This confirms – although less dramatically than with the no-progress time – that TCP's performance in an ad hoc network starts to seriously degrade after 2 hops. Note that the left figure has another scale and that with link layer feedback we have litterally no unfairness in this region.

3.4 Using OLSR instead of AODV

In order to exclude that AODV's on-demand routing style is responsible for the observed TCP stalls and unfairness, we ran the simulations using the proactive routing protocol OLSR.

In figure 4 we compare the TCP unfairness ratio when link layer feedback is disabled for both AODV (left figure) and OLSR (right figure). As can be seen, OLSR performs even worse and does not help to expand the ad hoc horizon.

4 Analysis and Discussion

The ad hoc horizon identified in the previous section is mostly due to TCP's behavior, as we will discuss in this section. Although TCP unfairness over IEEE 802.11 is a well known problem (and

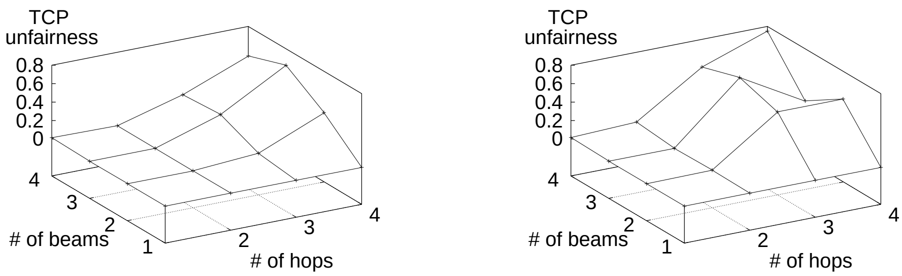

Figure 5: TCP unfairness without link layer feedback, comparing AODV and OLSR.

we do not add new insights here), we examine more closely why TCP unfairness occurs in the beamstar scenarios. This will permit us to draw some conclusions on where modifications are required for the triple of TCP, ad hoc routing and the 802.11 MAC protocol.

4.1 Packet-Level Analysis of TCP Behavior

A simple but nevertheless interesting scenario is the beam star topology with 2 beams and 3 hops. Figure 6 shows the communication pattern for two FTP sessions in that scenario where node N0 is the center node: For each of the beams an FTP session is started towards beams' head nodes N3 and N6. The figure shows the first 100 seconds of a simulation run with dark areas indicating data traffic while the dashed area is broadcast traffic due to AODV's control messages.

We see that initially both FTP flows can transfer some data from the center to their respective beam heads, but that repeated timeouts for the flow to the right leads to a rather large no-progress period of approximately 40 seconds! Later on, the flow to the right is able to become active again, but at the price of starving the left FTP flow.

Regarding the no-progress and the unfairness metrics plotted in section 3 this means that although the fairness is not too bad for this simulation run, we have a serious no-progress problem.

For understanding why the right FTP session fails to become active again after a first timeout, we show in Figure 7 a detailed view of the packet exchanges around the time when the first loss of a TCP acknowledgment occurs, as they were extracted from the simulation traces.

In the figure we have indicated "bow waves" with dashed lines. They represent the interference area of a speeding "forwarding train": As a packet is forwarded along a beam, it disturbes riple like all communications around its transmission place. The bow wave labeled with (1) is due to the forwarding of a data packet from the center node to the leftmost beam head. The internode distance is set to 130 m which means that the interference range of 550 m covers 4 hops. During the time that the left beam's bow wave is covering most of the right beam, the right side cannot even do a single RTS: Event (a) shows such an unsuccessful attempt.

When the activity momentarily stops at the left side, the right side manages to complete the forwarding of a data packet to the right most beam head. During this time, the bow wave labeled with (2) prevents the left side to proceed. Unfortunately for the right side, the left side regains control thanks to the persistence of the RTS/CTS scheme, where a new bow wave (3) starts immediately. This time the second-most left node is draining its packet queue towards the node N6: The previous bow wave also hindered the forwarding inside the left beam. Due to this massive

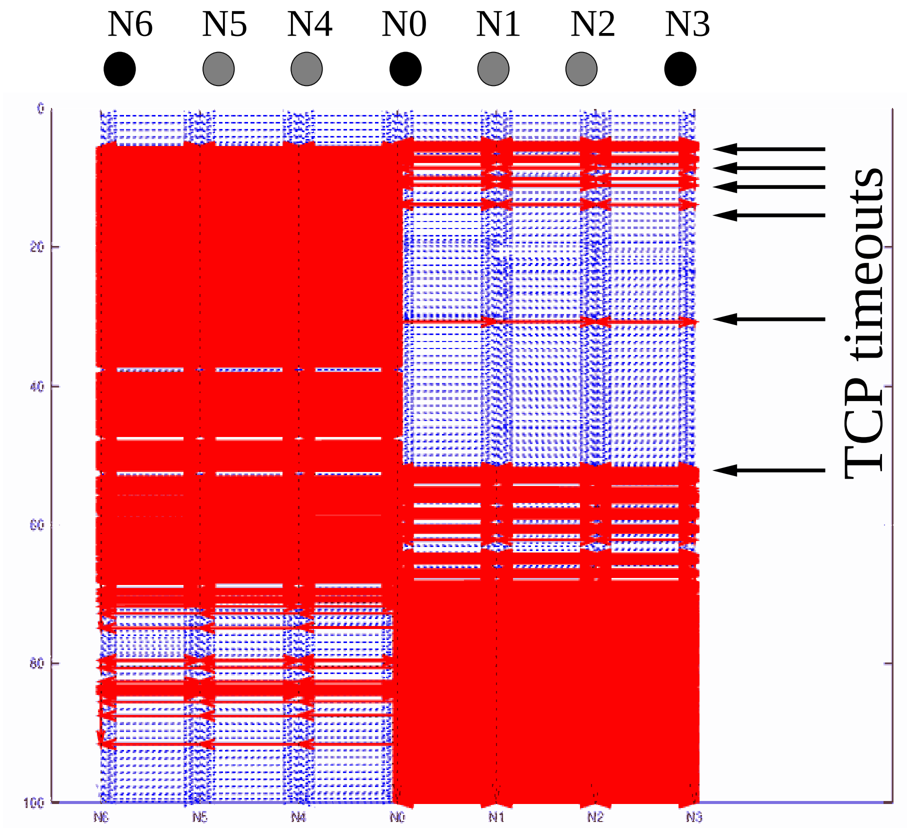

Figure 6: Alternating starving of the two beams' TCP sessions, leading to a low unfairness index but a high no-progress ratio.

blocking of the medium (which happens outside of the RTS/CTS coordination range for the waiting nodes on the right side), 3 RTS failures occur between event (c) and (d), eventually destroying the TCP ACK packet at the right most node that waited for expedition to the center.

4.2 Optimizing for Bad TCP Performance

After having seen the reach of the interference zone for a simple 3-beam-2-hop scenario, we wondered to which extent the distance between nodes would affect our ad hoc horizon result. We used the same 3-beam-2-hop scenario as above and increased the internode distance from the initial 130 m to the transmission limit of IEEE 802.11 in ns-2 that is, 250 m.

As can be seen in Figure 8, an internode distance of 150 m would produce even worse figures and thus yield a more marked ad hoc horizon. Whether this 150 m distance would also have been "optimal" for other beam stars has to be explored. Note that even when we had chosen an internode distance of 200 or more meters, the no-progress time ratio would have landed in the 20 to 30% range, which is beyond userfriendlyness. Moreover, in reality we will have heavily fluctuating link quality at these distances and could not count on such "good" TCP performances.

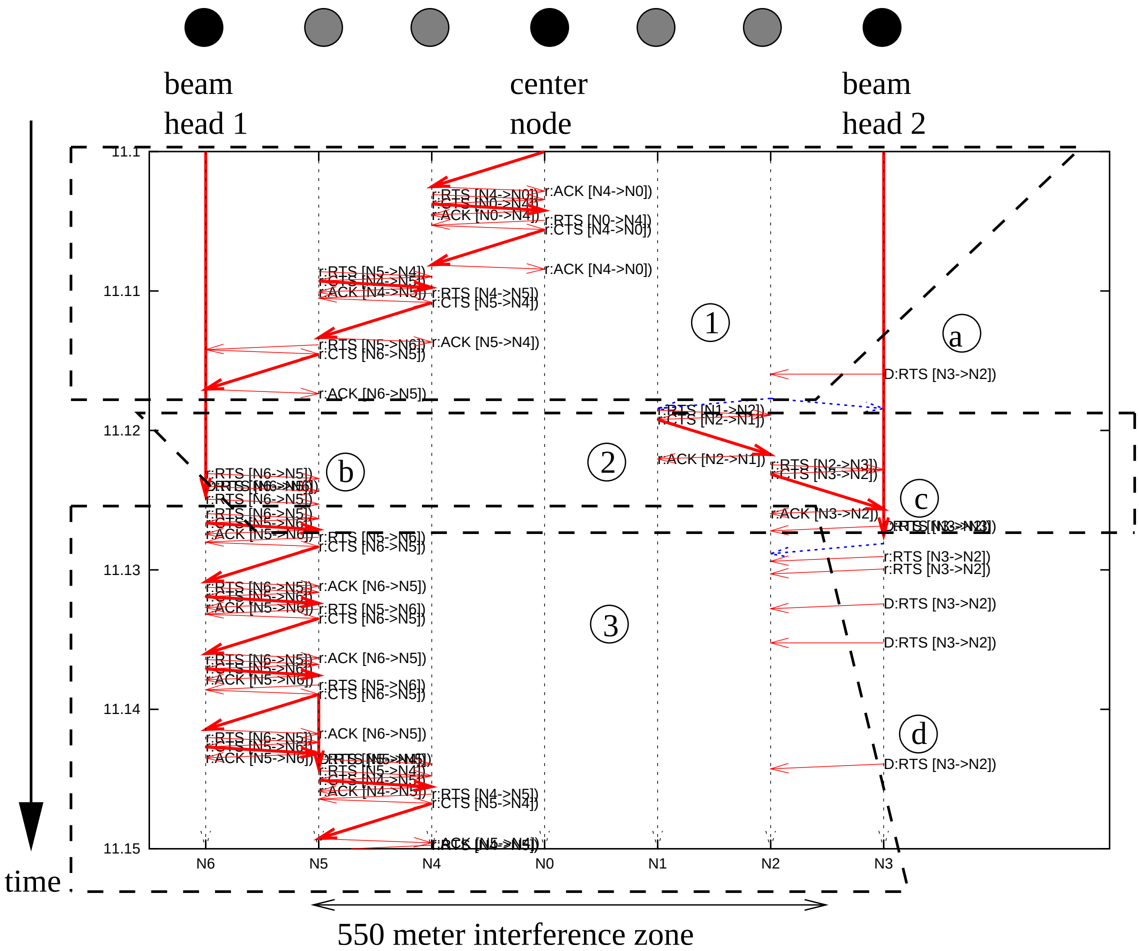

Figure 7: The seed of unfairness: RTS/CTS failure due to remote transmission activity.

4.3 Need for Cross Layer Optimization

A number of solutions have been proposed to remedy TCP's bad performance in a wireless context: Modifying MAC level scheduling [9], implementing distributed queue management [8] as well as other TCP modifi cations in order to adapt to the specifi cs of wireless multihop networks [2].

Although a thorough discussion of the proposed improvements to TCP are out of scope for this report, we fi nd the 3-beam-2-hop example useful to show that some of these proposals will not eliminate the problem of long TCP starvation.

MAC layer scheduling [9] indeed improves the *fairness* of the two TCP flows in scenarios equivalent to 2 beams and 2 hops. However, as discussed above, good fairness fi gures might co-exist with long no-progress intervals. Moreover, we have shown that the severe problem with no-progress time occurs mainly due to interferences with remote communications and is not due to collisions with other flows. We therefore suggest that changes in the way the MAC layer schedules its packets alone will not be able to reduce TCP's long no-progress intervals.

Another technique suggested for enhancing TCP fairness in ad hoc networks is distributed neighborhood RED [8]. The idea behind nRED is to coordinate the dropping of messages in

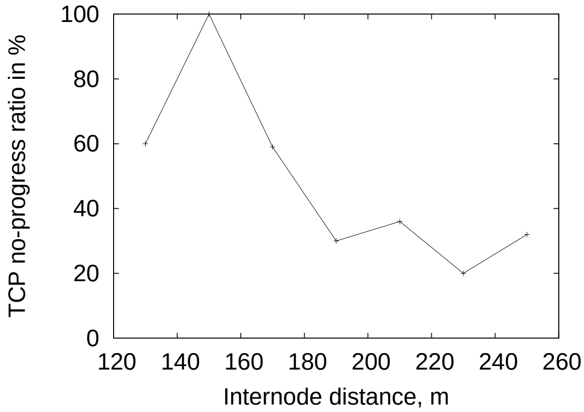

Figure 8: How changing the internode distance affects the no-progress metric for a 3-beam-2-hop scenario.

wireless nodes by overhearing the transmission in the neighborhood. As one can see in Figure 7, the loss of messages in the beam-star scenarios happen on the border of the interference zones of opposit communicating nodes where the signal is not legible. When message losses happen close to the center, the MAC layer can handle collisions properly *and* will be able to overhear neighborhood communications. In the beamstar topology, however, the relevant neighbors are too remote in order to let nRED learn about the details of the access collissions.

We conjecture that only a cross layer coordination between transport and MAC layer will permit to avoid situations like the one described earlier in this section. A TCP-agnostic MAC layer cannot prevent TCP from going too early into long backoff periods, while letting TCP learn from failed transmission attempts only is too little coordination which should be replaced by a more tight coupeling between MAC and TCP layer scheduling.

5 Conclusions

Although making optimistic assumptions on the environment and configuration for a wireless ad hoc routing protocol, we found by simulation rather discouraging worst case figures for the combination of plain TCP, AODV and IEEE 802.11. They confirm experimental insights that operating multihop forwarding is only recommended in a very limited range. This “ad hoc horizon”, as we call it, is in the low 2 to 3 hops range and extends to a dozen nodes.

While this ad hoc horizon depedends on the various protocols involved, and thus can potentially by extended by advances in cross layer coordination, we would be surprised if the horizon could be fully resolved. Even if the region inside ad hoc horizon would double, one will be restricted to rather small size multihop networks. For the forseeable future it is therefore advisable to optimize ad hoc routing protocols for their small-scale behavior rather than their properties when scaling up.

References

[1] AODV-UU implementation, Uppsala University, 2004. `http://www.docs.uu.se/scanet/aodv/`.

[2] E. Altman, and T. Jimenez, *"Novel delayed ACK techniques for improving TCP performance in multihop wireless networks"*, In Proc. Personal Wireless Communications (PWC 2003), Venice, Italy, 2003.

[3] J. Broch, D.A. Maltz D.B. Johnson, Y-C.Hu, and J. Jetcheva. *"A performance comparison of multi-hop wireless ad hoc network routing protocols"*, In Proc. MobiCom'98, Dallas, Texas, USA, 1998.

[4] M. Gerla, K. Tang, and R. Bagrodia. TCP performance in wireless multi-hop networks. In *Proc. of IEEE WMCSA'99, (New Orleans, LA).*, Feb 1999.

[5] G. Holland, N.H. Vaidya, *"Analysis of TCP performance over over mobile ad hoc networks"*, In Proc. MobiCom'99, 1999.

[6] S-J. Lee, E.M. Belding-Royer, and C.E. Perkins, *"Scalability study of the ad hoc on-demand distance vector routing protocol"*, International Journal of Network Management, Volume 13, Issue 2, March/April 2003, pp. 97 - 114.

[7] H. Lundgren, D. Lundberg, J. Nielsen, E. Nordström, and C. Tschudin, *"A large-scale testbed for reproducible ad hoc protocol evaluations"*, In Proc. *3rd annual IEEE Wireless Communications and Networking Conference (WCNC 2002)*, pages 412–418, Mar 2002.

[8] K. Xu, M. Gerla, L. Qi, and Y. Shu, *"Enhancing TCP fairness in ad hoc wireless networks using neighborhood RED"* , In Proc. MobiHoc'03, Annapolis, Mariland, USA, 2003.

[9] L. Yang, W.K.G. Seah, and Q. Yin, *"Improving fairness among TCP flows crossing wireless and wired networks"*, In Proc. MobiHoc'03, Annapolis, Mariland, USA, 2003.