Empirical Upper Bound on TCP Transmission Rate for Guaranteed Capture-Free Communications in Multi-hop IEEE 802.11 Based Wireless Networks

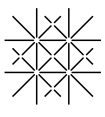
Evgeny Osipov

Technical Report CS-2005-001
University of Basel
February 8, 2005

Abstract

In this paper we consider the well known problem of TCP capture in mobile ad hoc networks, which is an unfair distribution of network bandwidth between TCP flows. The conventional way of analysis of TCP performance in wireless networks is to consider multiple flows in a network with perfect connectivity. In the majority of existing approaches that target solution to the TCP capture problem the effect of long range interferences is neglected. In this paper we look at the problem of TCP capture from a different prospective. We assume that all nodes of competing TCP flows are located outside the assured reception range of each other but within the physical carrier sensing range. We compute the bound on transmission rates at sources of TCP flows which guarantees capture free communications and close to perfect fairness between all involved flows. Our solution differs from all related approaches in the way that it does not modify the existing standards of the IEEE 802.11 MAC and TCP and fully complies with the end-to-end paradigm of TCP.

Keywords:
Ad Hoc networks, TCP capture, TCP performance,TCP fairness, IEEE 802.11.



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

In this paper we consider the well known and yet unsolved problem of TCP capture in mobile ad hoc networks (MANETs), which is an unfair distribution of network bandwidth between TCP flows. As a result, only few connections obtain access to the network's transmission capacity, severely degrading the performance of other flows.

The conventional way of analysis of TCP performance in multi-hop wireless networks is to consider multiple flows in a network with perfect connectivity. In all related work on improving TCP performance in such networks, the solutions which target the TCP capture problem suggest distributed algorithms to control the congestion [1], [2], [3]. The effect of long range interferences in the majority of existing approaches is neglected. Moreover all the related approaches target either modification of TCP mechanisms in order to make the protocol more "wireless friendly" [6], [7], [8] or modification of MAC 802.11 [1], [2], [3], [5]. In the light of the wide deployment base of the IEEE 802.11 products and popularity of the existing TCP/IP protocol stack, we foresee that these kinds of solutions will face considerable deployment problems.

In this paper we look at the problem of TCP capture from a different prospective. We assume that all nodes forwarding the traffic of competing TCP flows are located outside of the assured reception range of each other and find a solution that ensures capture free communications for multiple TCP flows without modifications to the IEEE 802.11 MAC or TCP protocol.

The major reason for consideration of a network without perfect connectivity between the nodes forwarding traffic of competing TCP flows is the spontaneous nature of ad-hoc communications where administrative frequency planning is not possible. Therefore we can assume that all nodes will communicate over the same radio channel. Thus interferences from some stations will definitely disturb the communication of other stations. As we will show in the paper this leads to severe unfairness between TCP flows.

1.1 Contribution and structure of the paper

To the best of our knowledge the study of TCP fairness in presence of long range interferences presented in this paper is unique.

Our major challenge is to answer the question: What is a common behavior for all traffic sources that assures optimal fairness between multiple TCP flows in a network where protocol based methods are unable to control the congestion? We compute the upper bound on transmission rate from the interface queue on the link layer at sources of TCP flows that satisfies this condition.

In our approach we focus on a solution which does not modify the existing standards of the IEEE 802.11 and TCP. Moreover our solution fully complies with the end-to-end paradigm of TCP since the obtained bound is used only to set the parameters of the scheduler of the interface queue on the link layer at sources of communications. Thus no changes to the forwarding nodes are needed.

The rest of the paper is organized as follows. We first introduce the problem of TCP capture and review the existing approaches aiming at its solution. In the same section we present the motivation for this work. After that in Section 3 we explain our methodology for derivation of the upper bound and outline our solution. In Section 4 we present the essential definitions and assumptions for this work. We elaborate properties of the newly defined terms in Section 5. The actual derivation of the upper bound follows in Section 6. We analyze and discuss the implications of the obtained bound further in Section 7 before concluding with Section 8.

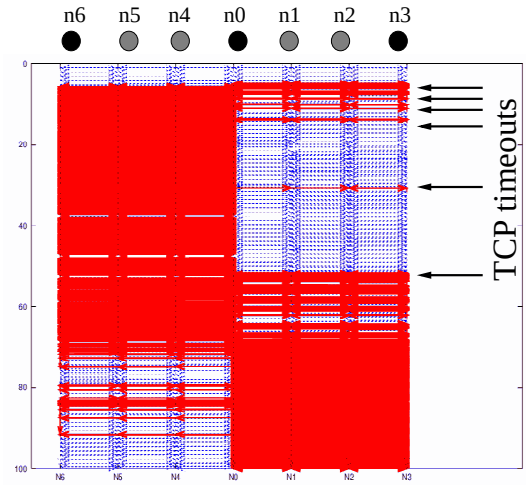


Figure 1: Alternating starving of the two FTP sessions leading to the TCP capture problem.

2 Related Work on TCP Capture Problem in Ad hoc Networks and Motivation

The problem of TCP capture is initially described in [11]. The core of the problem is consequent expiration of the exponential TCP retransmission timer for one flow due to congestion created by transmissions from nodes forwarding packets of other flows. The winning TCP flows utilize this temporal weakness of the affected flow by increasing their transmission rates. This behavior aggravates the situation for the affected flow. The best way to show the effect of the TCP capture problem is to illustrate it on a simple example.

2.1 The problem of TCP capture

In order to illustrate the problem of TCP capture we perform a simple experiment with only two TCP flows. The topology for the experiment is depicted in the upper part of Figure 1. The figure shows the communication pattern for two FTP sessions both starting in the central node $n0$ and following the path of three hops towards nodes $n3$ and $n6$ respectively. The figure shows the first 100 seconds of a simulation run in NS-2 [15] with dark areas indicating data traffic while the dashed area represents broadcast traffic due to control messages of a routing protocol. In this example we assume RTS/CTS exchange prior to data transmission. In section 6.1.1 we show the TCP capture problem in action in the case where RTS/CTS handshake is disabled.

We see that initially both FTP flows can transfer some data from the center to their respective destinations, 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.

For understanding why the right FTP session fails to become active again after a first timeout, we show in Figure 2 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 Figure 2 we have indicated “bow waves” with dashed lines. They represent the interference

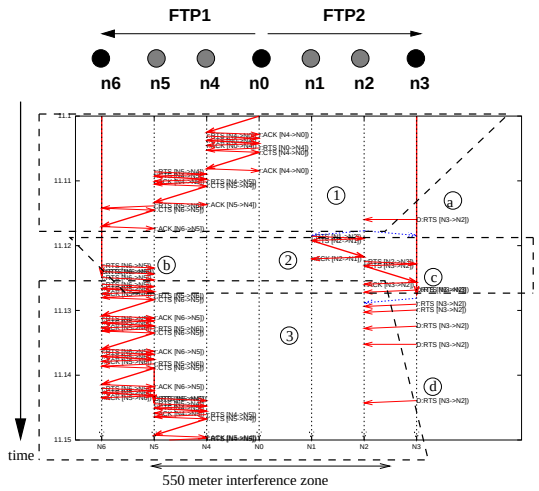


Figure 2: The seed of unfairness: Inability of communication due to remote transmission activity.

area of a speeding "forwarding train": As a packet is forwarded along a route, it disturbs 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 node $n6$. The internode distance is set to 126 m which means that the interference range of 550 m covers 4 hops. During the time that the left connection's bow wave is covering most of connection FTP2, the nodes on the right side cannot even complete a single RTS/CTS exchange: 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 node $n3$. During this time, the bow wave labeled with (2) prevents the left side to proceed. Unfortunately for FTP2, flow FTP1 regains control thanks to the persistence of the medium access method of the IEEE 802.11 MAC, where a new bow wave (3) starts immediately. This time node $n5$ is draining its packet queue towards the node $n6$: The previous bow wave also hindered the forwarding inside FTP1. Due to this massive blocking of the medium (which happens outside of the range of assured data communications for the waiting nodes on the right side), 7 RTS failures occur between event (a) and (d), eventually destroying the TCP ACK packet at the right most node that waited for expedition to the center.

2.2 Related Work on Improving TCP Performance

One line of work which indirectly deals with the problem of TCP capture attempts to improve TCP congestion control mechanism in order to increase its performance in wireless networks [6], [7] and [8]. The common point of these works is that they target TCP modifications in order to make the protocol more "wireless friendly". However as it is shown in [12] increasing TCP throughput as an ultimate goal of the network optimization problem leads to severe unfairness between the flows. The authors argue as well that the focus should be placed on fairness issues.

Another work of interest is [5]. The authors describe the approach of probabilistic hopping between available radio channels for all nodes communicating in the ad-hoc mode. The major idea is to announce the transmission schedule over different channels to all nodes in the neighborhood of one hop. It is shown that with such scheme the effect of interferences is minimized and the total throughput of the network is increased. However the major goal of our work is to understand

natural limitations of the IEEE 802.11 technology without modifying the functionality of the MAC layer. Therefore we do not consider similar approaches in the scope of this paper.

The number of attempts to actually solve the problem of TCP capture were undertaken during last several years. All together they can be characterized as an attempt to bring the known quality of service mechanisms from the wireline Internet to the wireless domain. The following works are the major representatives of this group.

The work in [2] suggests a distributed algorithm for exchanging the status of local queues and scheduling information between all nodes in the neighborhood of one hop. For this purpose the authors suggest modifications to the distributed coordination function to implement broadcasting of such information. Further on this information is used to compute relative weights for all packets waiting for transmission in all stations in a manner that mimics the behavior of the weighted fair queuing scheduling in the wireline Internet.

Another technique suggested for enhancing TCP fairness in ad hoc networks is distributed neighborhood RED [3]. The idea behind *nRED* is to coordinate the dropping of messages in wireless nodes by overhearing the transmission in the neighborhood of two hops from each node and estimating the size of a virtual queue of the whole neighborhood.

Finally amongst the approaches targeting the solution of the TCP capture problem we distinguish the work in [4] as it has common parts with the approach described in this paper. The authors propose to limit the arrival rate from the interface queue to the transmission buffer on MAC layer depending on the observations of the departure rate in the past. The common point of this work to our approach is that the solution assumes rate limitation at a transmitting node implemented on the link layer, thus leaving the functionality of the MAC protocol unchanged. However this approach differs from ours in several aspects. Firstly this solution assumes rate throttling in forwarding nodes as well as in sources of communications while in our approach we limit the rate at sources of TCP flows only leaving the functionality of the forwarding nodes unchanged. Secondly the rate limits in [4] are computed based on heuristics while for computation of our rate bound we use actual properties of TCP communications. Finally this approach was tested in simple scenarios while we consider the performance of our solution in the worst case.

2.3 Motivation

In [14] it is shown that the reason for TCP capture to appear is the destructing effect of long range interferences. The authors show that the loss of messages which lead to consequent expiration of TCP timeouts for one flow happen on the border of the interference zones of communicating nodes that are located outside the communication zone for the affected node. This behavior is also illustrated in Figure 2. This means that solutions built on the protocol based exchange of information between the nodes in the network simply do not work in this case.

Since the problem appears outside the communication zone of the affected node the solutions targeting modification of TCP congestion control mechanisms would not work either. In this case the notion of a bottleneck link/zone is somewhat different from the conventional definition in the wireline Internet. In general as we show in this paper the TCP flows which create the congestion for the affected flow do not feel the congestion at all. As soon as the throughput of one of the flows goes to zero the flows that caused the situation continue to fill the spare capacity.

This motivated us to consider the case where there is no way for the punished TCP flow to affect the behavior of the competitors.

3 Methodology

This section describes our approach for analyzing the behavior of TCP flows that lead to appearance of the TCP capture problem, hence to severe unfairness, when flows are located outside the communication areas of each other.

Amongst all TCP flows we differentiate two kinds of them: Those which cause the TCP capture we will refer to as the *interfering* flows and those which suffer from the TCP capture we will refer to as the *affected* flows. On a general network topology there can be several flows of each kind. The core of our methodology is to analyze the conditions for several interfering flows which cause appearance of the TCP capture problem for a single flow and then generalize the solution. Throughout the text we will refer the single affected TCP flow as the *sample* flow.

Overall our approach can be characterized as "from a particular to general". We build our reasoning first by considering the case where all flows span over a path of maximum length. The major reason for this is to observe the conditions for the TCP capture to appear in the case where the opportunities of each flow are equal in terms of TCP settings and the path length. In the case where interfering flows have shorter routes they have an additional advantage with respect to our sample flow. In Section 7.2 we relax this assumption and consider TCP flows with shorter routes.

The actual research line is divided into two stages:

1. Observation of the properties of a single isolated and several interfering TCP flows.

2. Analysis of previously observed properties in order to understand a common behavior of all interfering flows that prevent TCP capture for the sample flow.

On the first stage we analyze the behavior of a single isolated and several multi-hop TCP flows separated by distances larger than the range of assured data reception. As the result of the analysis we extract properties of TCP communications in multi-hop wireless networks with respect to the level of interferences produced by all nodes transmitting their traffic. In particular we derive the worst case placement of the interfering TCP flows in space such that the effect of interferences produced by transmissions from these flows on the performance of the sample TCP flow is worst.

In order to extract these properties we introduce new definitions of communication ranges (Section 4) for the entire TCP flow. These newly defined terms allow us to prepare a theoretical background for the analysis of the interference properties of one and several multi-hop TCP flows. The analysis itself is done based on series of simulations. The last two steps are described in Section 5.

The major outcome of the first stage is the definition of the worst case position of the interfering TCP flows with respect to the sample flow affected by the transmission activities of these flows. On the second stage we utilize this result for the construction of the worst case scenario with respect to the sample flow. In the constructed scenario we first show that without any changes the interfering TCP flows are able to completely shut down the transmission of our sample flow. Then we observe the case where under worst placement of two TCP flows the problem of TCP capture does not appear. This allows us to define *optimal* level of fairness in the worst case. We extrapolate this result for multiple interfering flows case and derive the major result of this work: the bound on transmission rate at sources of TCP flows. We use the obtained bound to configure the scheduler of the local interface queue on the link layer of TCP senders which which guarantees capture free communications for all competing flows. All the outlined steps are elaborated in Section 6.

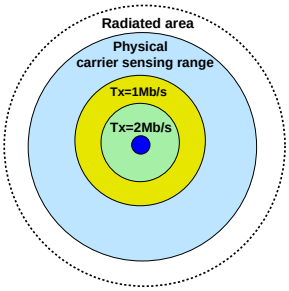


Figure 3: Communication ranges of the *base* IEEE 802.11 devices.

In our solution the bound is a function of: (1) Total transmission rate of single isolated multi-hop TCP flow; (2) The number of nodes transmitting the TCP data segments of an interfering connection; (3) The number of interfering connections under the worst case placement. The first parameter is estimated during the first phase of our investigations. The second parameter is obtainable from a reactive ad-hoc routing protocol as described in Section 7.3, for the experimental evaluation we configured this parameter in advance since we have a perfect knowledge about the flows in the constructed topologies. Finally the third parameter is derived as the result of the analysis of the worst case topology.

4 Definitions and assumptions

In the main focus of this paper is the behavior of an entire multi-hop TCP flow. Further on in the text we will refer a set of nodes including the source, destination and nodes that forward packets of the TCP flow as a *connection*. Thus the term *flow* includes all the mechanisms and characteristics of the TCP communications such as slow start, congestion avoidance, throughput, bandwidth delay product, etc. The following subsections define characteristics of the term *connection*.

4.1 New definitions of radio transmission ranges of a connection

Conventionally when talking about transmission ranges of the IEEE 802.11 enabled devices a single transmitting node is in focus. For the single node the radio transmission ranges are defined as schematically shown in Figure 3.

In order to facilitate readability of the paper we denote the area of assured data reception up to the slowest transmission rate (1 Mb/s) as *MAC zone* of a node. Denote as well the area beyond the MAC zone but within the physical carrier sensing range as β*MAC zone* (beyond MAC) of a node.

In this paper we extend the definitions of the transmission ranges for the entire multi-hop *connection* and define MAC region, βMAC region and the area of maximum interference as follows.

MAC region is a space around the connection in which other station not included in the connection are able to receive data packets issued by the connection with the basic data rate of 1 Mb/s.

βMAC region is a space around the connection beyond the MAC region of the connection but within the βMAC zone of every node included in the connection.

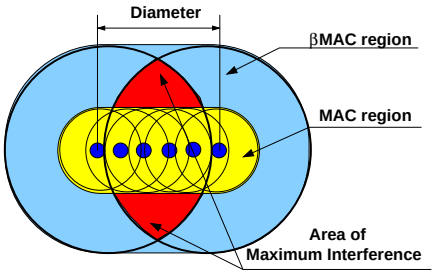


Figure 4: Communication ranges for the entire connection (internode distance 126 m).

Nodes of other connections located in this area can not receive data packets from this connection but reception of own packets are affected by interferences produced by nodes of this connection. The characteristic of the βMAC region is *interference load*.

Interference load is a measure of the activity of the connection on the physical layer in the βMAC region.

In general the interference load is the characteristic of a signal on the physical layer and cannot be measured in bits per second since no data communication is possible. However obviously there is a correspondence between the data rate observed on the MAC layer at senders of data packets and the characteristics of the signal in the βMAC region. Since this paper does not deal with models for radio transmissions of the IEEE 802.11 technology we will measure the interference load as total data transmission rate in bits per second observed on MAC layer of sending nodes of the connection.

Area of maximum interference is the area of the βMAC region of the connection where maximum interference load is observed. Further on we will refer to this term as *AMI*.

Diameter of the connection is the linear distance between two most distant nodes of the connection.

Figure 4 illustrates the above definitions on the example of a five hops connection with internode distance equals 126 m.

For further discussion we define as well the following terms which are conceptually shown in Figure 5.

Epicenter is the intersection of the areas of maximum interference of two connections.

Focus of epicenter is the line between the two most distant points of the epicenter.

Section 5 elaborates the properties of the area of maximum interference, epicenter and focus of epicenter and we proceed with the assumptions which are essential for this work.

4.2 Assumptions

In Section 2.3 we argued the need to consider the connections separated by interference distances. Therefore the major assumption is that no one node of one connection is located in the MAC region of other connections.

We assume there exists another solution dealing with the case where several flows in the βMAC region of the sample flow share the MAC regions of each other. The processes ongoing during communications of several TCP connections partly or completely within the MAC region of each other are different and falls beyond the scope of this paper. Therefore we leave the

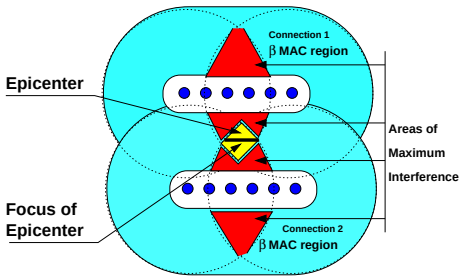


Figure 5: Epicenter and focus of epicenter.

full analysis of such mechanisms for our future investigations and present our view on what these mechanisms should be in Section 7.

4.2.1 Radio Transmission model and sizes of communication zones for a node

For the worst case analysis we assume radio transmission model of the original IEEE 802.11 standard used in NS-2 [15]. Therefore all values for sizes of respective MAC/βMAC zone of a node, MAC/βMAC region, area of maximum interference, focus of epicenter of a connection are specific only for this model. The reason for using NS-2 radio transmission model is that it gives us an opportunity to work deterministically with the transmission ranges, which we need for construction of the worst case scenario. Obviously using other models as well as precise measurement of communication regions in real networks would result in different values of the communication distances. However as we will show in Section 7 discussing the practical applications of the obtained bound this assumption does not affect the validity of the results in this paper.

Using the parameters of 914 MHz Lucent WaveLAN DSSS radio interface, the radius of MAC zone of a node is $r_{MACzone} = 250$ m and the radius of the βMAC zone is $r_{\beta MACzone} = 550$ m. Further on we define the βMAC zone of the node as a region limited by the internal circle of radius $250 + \delta$ m and the external circle of radius 550 m. We assume that on a radius of $550 + \delta$ m the interference produced by the node is zero. For simplicity of simulations setup we define $\delta = 1$ m.

We define the minimum internode distance (d_{min}) for a multi-hop hop connection the distance which satisfies the condition

$$d_{min}(N_i, N_{i+1}) = \frac{r_{MACzone}}{2} + \delta. \tag{1}$$

The minimum internode distance condition (1) dictates that any two nodes N_i and N_{i+2} communicating through another node N_{i+1} (a two hops communication) should be located just outside the MAC zones of each other. Therefore in order to satisfy this condition the intermediate node N_{i+1} should be located half way between N_i and N_{i+2}.

We define the maximum internode distance (d_{max}) for a multi-hop hop connection the distance which satisfy the condition

$$d_{max}(N_i, N_{i+1}) = r_{MACzone}. \tag{2}$$

The maximum internode distance condition (2) means that any two neighboring nodes should be located at the border of MAC zones of each other.

Thus for a straight line topology $d_{min} = 125 + \delta = 126$ m (depicted in Figure 4) and $d_{max} = 250$ m.

4.2.2 Three dimensional case

In order to facilitate the analysis and extract the essential properties of TCP communications in multi-hop wireless networks we consider only two dimensional scenarios. Our methodology is naturally extensible for the three dimensional case however we leave the development of this issue for our future work.

4.2.3 Path establishment and length of routes

Since forwarding of routing messages introduces additional load to the network in order to simplify the analysis we assume that all routes are stable for the whole duration of experiments.

For the experimental evaluation of the results we use the Lightweight Underlay Network Ad hoc (LUNAR) [16] routing protocol. LUNAR was designed as a simpler alternative to the IP level ad-hoc routing protocols. The major characteristics of this protocol relevant to the topic of this paper are its place in TCP/IP protocol stack and the control over total size of the ad-hoc network. LUNAR is located at level 2.5 namely above the link layer and below IP. The second characteristic embedded in LUNAR is the controllable limit on maximum number of hops which flow can traverse.

The reason for choosing LUNAR for the evaluation of our ideas is the simplicity of control on the behavior of the interface queue which offers its implementation for NS-2 and possibility to control the length of established paths.

As it was shown in [14] the severe TCP capture problem appears already with two TCP flows spanning over three hops and does not occur in two hops case. We account for this result in this work. Moreover we decided to increase the maximum length of the routes allowed in the network more than two times in comparison to the case where TCP capture does not appear. Therefore we enforce a 5 hops limit on maximum number of hops for a TCP connection.

Even though all results in this paper are obtained from the experiments on static topologies we do not explicitly exclude the mobility. The nodes of all connections are free to move with the only condition that nodes of one connection do not fall in the MAC region of another.

4.2.4 Traffic Characteristics

In this paper we consider only communications over TCP protocol. We consider TCP Newreno as the most popular variant of TCP. It was shown in [9] that TCP achieves highest throughput in a multi-hop wireless network when the congestion window size is scaled at sources as $h/4$, where h is the number of hops traversed by the flow. Further studies of the effect of TCP parameters on the end-to-end performance described in [13] refine this bound and show that with window size equals $3 \cdot h/2$ TCP achieves optimal throughput on a general topology. For the experimental evaluation of our solution we decided to account for this result and set the window size to 7.

In the same source a careful analysis of the effect of RTS/CTS exchange on TCP throughput in multi-hop case is given. It is shown that on all topologies disabling the RTS/CTS handshake improves the throughput of TCP. We account as well for this result and disable RTS/CTS exchange in all simulations.

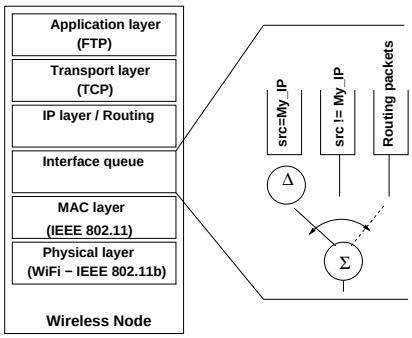


Figure 6: Model of the IEEE 802.11 enabled node.

4.2.5 Model of a wireless node

Schematically the architecture of a wireless node is presented in Figure 6. We assume the following structure of the scheduler of the interface queue on the link layer. We assume the interface queue either logically or physically divided into three queues: One for data packets originated at this node, the second one for all other data packets and the third for all packets of the routing protocol. All three queues are drop-tail in nature. Upon arrival from the routing layer all packets are classified according to their source IP addresses and type and placed into the corresponding queue.

The scheduler from the interface queue to MAC layer consists of two stages. On the first stage we have a fixed delay non-work-conserving scheduler with tunable delay parameter Δ. When $\Delta = 0$ the first stage scheduler works as usual work-conserving scheduler and the whole scheduling system works as a simple scheduler with three priorities. The transmission rate bound derived in this paper will be used to set parameter Δ in all sources of TCP flows in order to guarantee capture free communications.

Scheduler Σ on the second stage is a non-preemptive work-conserving scheduler with highest priority to the queue with routing packets then to the queue with locally generated packets and finally to the forwarding queue.

4.3 Measurement Setup

The exact description of experiments follows in corresponding parts of the paper. Here we describe the common settings for all the experiments. All results presented in this paper were tested in NS-2 simulator of version 2.27.

The routes for all flows are established prior to the data transmissions. with interval 500 ms from each other which assures undisturbed path establishment for all flows. In all simulations the TCP flows traverse a paths from the most left node to the most right node. As all TCP flows started we allow a warm-up period of 5 seconds to exclude initial traffic fluctuations from the measurements. The duration of all simulations is 100 seconds. In order to produce statistically reliable measurements we repeat each simulation 30 times seeding randomly the RNG of NS-2 each time.

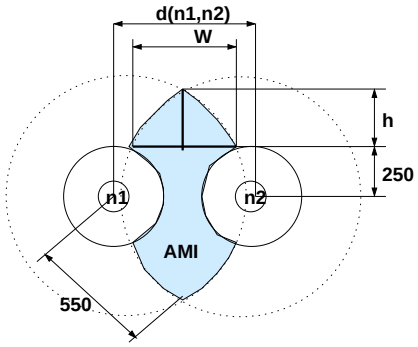


Figure 7: Area of maximum interference of two transmitting nodes.

5 Area of maximum interference, epicenter and focus of epicenter of multi-hop TCP connections

In this section we introduce properties of the communication regions of a connection defined in Section 3, motivate usage of string topologies for further analysis and analyze the interference load in the area of maximum interference.

5.1 Area of Maximum Interference

Let us consider two abstract transmitting nodes on distance $d(n1, n2)$ as shown in Figure 7 specifically let us concentrate on the βMAC zones of these two nodes. Obviously the process describing the signal in βMAC zone of a node is determined by transmission activity of the node. In the case of two transmitting nodes the signal in the area of intersection of their βMAC zones is determined by joint transmission activity of these nodes.

The purpose of introducing the term AMI is to answer the question: What is the length of a multi-hop connection that can be placed in the area with an interference load created by maximum number of nodes from another connection? Therefore the major dimensional characteristics of AMI is the width (*W*) and the height (*h*) of the area. The basic properties of *W* and *h* are observable from our two node example.

The width of the AMI is maximum when two nodes are located in the same geographical position. When the two nodes are located on a distance $d(n1, n2) = 2r_{\beta MACzone} + \delta$ the width of the AMI is zero.

In the two nodes example the height of the AMI is the distance between the tangent of the upper point of the MAC zone of a node and the point of intersection of βMAC zones of two nodes. As we shall see in general for a multi-hop connection the height of the AMI is determined by the tangent of the lower/upper point of MAC zones of the nodes with minimum/maximum coordinates along y-axis if x-axis is parallel to the diameter of the connection.

Consider a five hops connection of nodes placed on a straight line, in particular let us focus on the two extreme cases for node placement, the maximum distance and the minimum distance placements. The shape of the area of maximum interference obviously will be different for different internode distances. For the maximum distance node placement (2) it is enough to observe that AMI consists of four small areas (two AMIs located symmetrically above and below the diame-

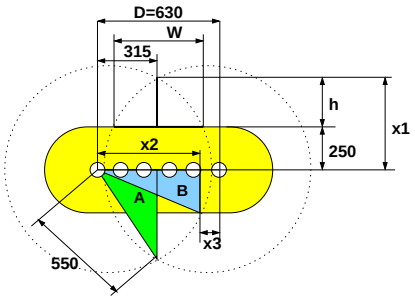


Figure 8: Width and height of AMI for the string topology with minimum internode distance.

ter of the connection) where the interference load is determined by transmission from five nodes. There is no region of such connection where the interference load is determined by transmission from six nodes. Verification of this observation is expelled from the paper due to its size and left to the reader.

The most interesting case for us is the minimum distance node placement on a straight line. As it is shown below the the interference load in AMI in this case is highest and determined by transmission from all six nodes of the connection.

Let us calculate the actual values for *W* and *h* parameters of the AMI for this case. Schematically the solution is shown in Figure 8. Note that in this case the two AMIs of the connection are symmetrical with respect to the diameter of the connection.

The height of AMI h is calculated as $h = x1 - 251$, where $x1$ is calculated from solving triangle A. The width of AMI W is calculated as $W = 630 - 2 \cdot x3$, where $x3 = 630 - x2$ and $x2$ is calculated from solving triangle B. In actual numbers for the string topology with internode distances 126 m the AMI has the following parameters: $h = 199.86$ m and $W = 348.78$ m. Thus on a line of the size equal to the width of AMI it is possible to place maximum three nodes to form a two hops connection with minimum internode distances and a three hops connection is not possible.

As we have already mentioned the width of the AMI is larger when the diameter of the connection is smaller. In Appendix A we show that further minimization of the diameter of the connection does not allow to accommodate more nodes in the largest part of AMI than on the string topology with minimum distance placement of nodes. Therefore for further analysis we will use only minimum distance string connections.

5.2 Maximum interference load of one and two connections

The major qualitative characteristic of AMI is the interference load. As we have learned above the maximum interference load occurs in a region of joint interferences from all nodes of the connection. For the string topology this happens when internode distances in the connection are minimal. In this subsection we experimentally analyze the interference load in AMI of one connection and in the epicenter of AMIs from two parallel such connections.

TCP segment size, B	Total interference load in AMI (b/s)		
	Mean	Median	Std. deviation
200	1371471	1371442	2361
400	1507020	1506972	3987
600	1592795	1592976	4216
800	1654868	1655270	4774

TCP segment size, B	Interference load from TCP data segments only (b/s)		
	Mean	Median	Std. deviation
200	810851	811093	1414
400	1045411	1045526	3104
600	1198417	1198901	3892
800	1307411	1307666	4259

Table 1: Interference load in AMI of one connection.

5.2.1 One TCP connection

As we discussed in Section 4.1 the interference load is determined by the joint transmission rate on MAC layer from all sources forming the area of maximum interference. In order to estimate this parameter in the case of one minimum distance (126 m) straight line connection we performed a series of experiments on such connection for different sizes of TCP data segment. For each value of the segment size we performed 30 simulation runs seeding randomly the RNG of NS-2 each time. We measured the average total transmission rate from all nodes of the connection on MAC layer. The resulting mean, median and standard deviation values are shown in the upper part of Table 1.

We observe a natural increase of the interference load with increase of the TCP segment size. In a loss-less case as we have here the TCP throughput, hence the interference load produced by the flow is larger for larger TCP data segment sizes. Note that for a TCP connection the interference load in the AMI is constituted from transmissions of two types of packets: TCP data segments and acknowledgments. For further discussion during each experiment we also measured the part of the total interference load in AMI contributed by TCP data segments only. The lower part of Table 1 shows the corresponding values.

5.2.2 Two parallel TCP connections

This time we evaluate the interference load in the epicenter of AMIs from two parallel minimum distance straight line connections. In order to estimate this parameter we perform a series of simulations with different linear distances between two connections. We vary positions of all nodes of the second connection along y-axis from 251 m (just outside the MAC region of the first connection) to 651m when two connections are located outside the βMAC regions of each other. For each value of the distance we performed 30 simulation runs seeding randomly the RNG of NS-2 each time. For all the experiments we use only one TCP segment size 800 B. We measured the total transmission rate from all nodes of two connections on MAC layer. The resulting mean,

Distance between connections, m	Interference load in epicenter (b/s)		
	Mean	Median	Std. deviation
251	1777932	1788672	34785
351	1770497	1793282	54748
451	1880221	1906665	92834
551	3312309	3313645	6231
651	3310974	3310004	6703
751	3308900	3310467	6091

Table 2: Interference load in epicenter of AMIs from two connections (packet size 800 B).

median and standard deviation values are shown in Table 2.

Observe that on distances 251 m, 351 m and 451 m our two connections are located in the βMAC regions of each other. This means that the transmission rate of one flow is affected by the transmission activities from another. This explains the smaller values of the joint transmission rate from all nodes of the two connections than on other three distances. Even more if we compare the values of the interference load for the first three distances with the interference load produced by one flow with the same packet size in Table 1 we observe that when two flows share their βMAC regions the total interference load is slightly larger than that of one isolated connection. As we observe later in Section 6.1 in this case there is some amount of unfairness in bandwidth distribution between the two flows. This also explains larger deviations of the interference load from the mean value on these three distances.

As for the other three distances we observe that the total interference load is double the value of the interference load produced by one flow. First time it happens when two flows are located just outside the βMAC regions of each other (551 m). The obvious explanation is that in this case the two flows behave completely independently, which is also reflected by small values of the standard deviation for these distances.

5.3 Epicenter and Focus of Epicenter

In Section 5.1 we argued that a minimum distance string connection forms the maximum AMI. Let us analyze the epicenter and the focus of epicenter of two parallel such connections. We begin with observation of the basic properties of epicenter.

1. The length of the focus achieved for two minimum distance straight line connections placed on the linear distance $d(TCP_1, TCP_2) = 2r_{\beta MACzone} + \delta$ from each other is zero.

2. The length of the focus achieved for two minimum distance straight line connections placed on the linear distance $d(TCP_1, TCP_2) = r_{MACzone} + \delta$ from each other is maximum.

Property one above is straight forward. Since βMAC regions of two connections do not intersect then obviously the size of the epicenter (hence focus) is minimum.

As for the property two even though the focus is largest in this case we learned from previous subsection that the interference load is slightly larger than that of one standing alone connection because of mutual effect of interferences. In the same section we observed as well that maximum interference load from two sessions is achieved when they are located at least on a linear distance

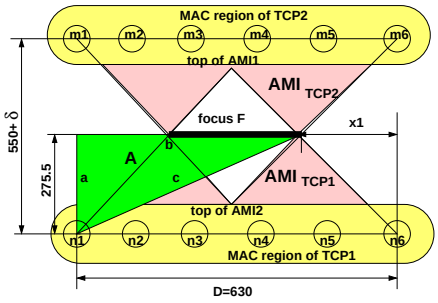


Figure 9: Worst case placement of interfering connections.

$d(TCP_1, TCP_2) = r_{\beta MACzone} + \delta$. In this case the interference load is the sum of loads from the two connections.

The last observation means that for the nodes of another connection that are located in the focus of the epicenter under such placement of the interfering connections the effect from the interferences is worst. Figure 9 illustrates this worst case placement of the two connections creating the largest interference load in the epicenter with graphical solution with respect to the length of the focus of epicenter.

The focus length F is computed as $F = D - 2 \cdot x1$, where D is the diameter of the connections (equals 630 m); $x1$ is computed as $x1 = D - b$, where b is the side of triangle *A* with two known sides a and c. Since the top of triangle *A* between b and c is the intersection point of the upper borders of βMAC zones of $n1$ and $m1$ (see also Figure 5 for clarification) and the radii of the border circles are equal then the line between two opposite intersection points halves the linear distance between the nodes $d(TCP_1, TCP_2) = 551$ which gives us the size of side $a = 275.5$ m. Side *c* is the radius of the outer border of the βMAC zone of node *n1* and equals 550 m. After calculations the resulting focus length $F = 322$ m and the focus begins at coordinates $(x_{n1} + 154, y_{n1} + 275.5)$.

6 Derivation of the bound for TCP transmission rates at sources

In this section using the above observations we derive the bound for TCP transmission rate at sources of the flows which guarantees capture free communication under worst placement of nodes of the interfering connections. First we construct the worst case scenario, illustrate the problem of TCP capture in action in this scenario and do final observation of properties of TCP connections. Finally we summarize all our knowledge, derive the bound and experimentally evaluate it.

Note that the performance metrics used in this section for evaluation of TCP performance are defined in Appendix A.

6.1 Step one: construction and analysis of the worst case scenario

With the parameters and coordinates of the focus computed in Section 5.3 we construct the worst case scenario as shown in Figure 10. There *TCP3* is our sample connection.

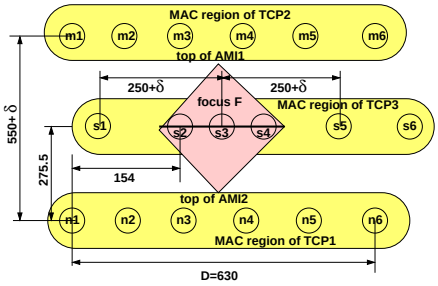


Figure 10: Worst case scenario for multi-hop interfering flows.

	Mean TCP throughput, kb/s			
Segment size, B	TCP1	TCP3	TCP2	Fairness index f_1
200	108	0.033	108	0.66
400	167	0.067	167	0.66
600	205	0.101	205	0.66
800	231	0.134	231	0.66

Table 3: Fairness of three TCP flows under worst case node placement.

6.1.1 Three flows

Let us show how bad TCP capture could be on the example of three TCP flows in the constructed worst case topology. Table 3 summarizes the results from 30 simulation runs with respect to the throughput achieved by each flow and the fairness index f_1 (see equation (5) in Appendix A).

As we can observe from all simulations, flow TCP3 always experiences severe TCP capture problem. In the majority of simulation runs it even did not have an opportunity to deliver a single packet. The reason for this is obvious: The interfering flows TCP1 and TCP2 are located outside the βMAC regions of each other and do not feel the congestion created in the area between them at all. Let us now analyze the situation where only two flows are competing under the worst case nodes placement.

6.1.2 Two flows

We have already studied two parallel connections located in the βMAC regions of each other in Section 5.2.2 with respect to the total interference load in the epicenter of their AMIs. Now in our focus the fairness properties of these connections. Let us analyze the case where three nodes of one connection (TCP3) are located in the AMI of another (TCP2). The topology for the experiments is similar to the worst case topology in Figure 10 but without flow TCP1 in this case. This time we measure the maximum TCP no-progress ratio and fairness index f_2 as defined by (6) in Appendix A for flows TCP2 and TCP3. Table 4 summarizes the results from 30 simulation runs.

In addition to the results presented in the table we also observed that in up to 50% of the runs the maximum TCP no-progress ratio was zero. As we observe TCP capture does not appear in this case however the throughput of flow TCP3 is always less than that of TCP2 ($f_2 > 1$). Obviously

Segment size, B	Max no-progress ratio, % Mean	Median	Std. deviation	Median of f_2
200	3.1419	3.2516	4.0759	2.3243
400	2.9630	0.0000	4.5506	1.9793
600	4.9265	3.2820	5.9751	1.7796
800	4.9093	1.6164	6.5055	1.7605

Table 4: Fairness of two TCP flows under worst case node placement.

Segment size (+overhead), B	Total load of TCP data segments, b/s	Rate limit at source, b/s	Delay at scheduler, ms
200 (+100)	810851	81085.1	29.59
400 (+100)	1045411	104541.1	38.26
600 (+100)	1198417	119841.7	46.72
800 (+100)	1307411	130741.1	55.07

Table 5: Parameters for throttling of TCP sources ($\alpha = 1$).

further increase of the interference load would result in decrease of the fairness and increase of the no-progress metric for connection TCP3 and finally lead to TCP capture.

We refer the level of fairness observed in the case of two parallel TCP connections under worst placement of nodes as *optimal fairness* [1].

6.2 Step two: the bound on transmission rates at sources of TCP connections

The way of computation of the upper bound in the worst case is somewhat straight forward if we summarize all previous observations and apply them to our worst case topology.

In order to prevent the TCP capture and achieve the *optimal fairness* amongst the competing flows in the worst case scenario the interference load in the epicenter of AMIs from TCP1 and TCP2 should be less or equal to the load produced by one flow. We already studied the interference load in the AMI of one connection in Section 5.2. Recall that the total interference load in AMI of one connection is $Load_{total} = load_{TCPdata} + load_{ACKs}$. Hence in order to mimic the behavior of one TCP flow the rate at sources of the two interfering connections should be limited according to (3).

$$Rate_{source} = \frac{\alpha \cdot load_{TCPdata}}{k \cdot n}. \tag{3}$$

In the above formula $\alpha \geq 1$ is the load correction multiplier specific to the fixed delay scheduler of the interface queue as explained below. The term n is the number of nodes of one connection that contribute to the maximum interference load from transmission of TCP data segments

[1] Of course one way of proceeding further is to find such a condition for TCP sources that the fairness between two flows would be *perfect* and not only *optimal*. However we consider the worst case situation as very unfortunate for flow TCP3 and assume that it is relatively rare event. Our solution already assumes reduction of the total throughput in the network. Therefore we see further reduction as not necessary and allow some level of unfairness.

Segment size,B	Median of total interference load (b/s) Single connection TCP1	Two shaped connections TCP1 and TCP2	Decrease in total load, %
200	1371442	1247956	9
400	1506972	1365436	9
600	1592976	1444207	9
800	1655270	1506031	9

Table 6: Decrease in total interference load due to shaping.

Segment size (+overhead), B	Load of TCP data segments, single flow, b/s	Correction multiplier, α	Delay at scheduler, ms
200 (+100)	810851	1.09	27.1
400 (+100)	1045411	1.09	35.1
600 (+100)	1198417	1.09	42.8
800 (+100)	1307411	1.09	50.5

Table 7: Final parameters for throttling of TCP sources.

in the epicenter of two AMIs. In the case of the minimum distance string topology $n = 5$. The multiplier k in the denominator is the number of connections between which we need to distribute the load. In the case of the worst case topology $k = 2$. Note that this rate limit include all MAC and IP level overheads to the data packets. Table 5 shows the rate limits and the delay parameter of the scheduler to the interface queue on the link layer at sources of TCP flows depending on the segment size used by the transport layer and $\alpha = 1$. The corresponding values for the part of the interference load produced only by TCP data segments of one isolated TCP connection are taken from Table 1.

Note that the delay values presented in Table 5 are computed based on the average values of load from data segments. Since we deploy a fixed delay scheduler in the interface queue at sources shaping the outgoing traffic with this delay parameter will result in CBR nature of the outgoing TCP traffic from the source. Therefore some loss in total load from the two interfering flows might be expected, Table 6 illustrates this discrepancy.

The nine per cent decrease in the load indicates that using the fixed delay scheduler in order for two connections to produce similar total interference as a single connection the outgoing rate of TCP data segments should be 1.09 times higher. Computing the delay parameter of the scheduler taking into account this correction to the total load produced by TCP data segments, our two interfering flows precisely model the total interference load of one flow. Table 7 shows the final delay parameters of the scheduler applying the correction multiplier.

6.3 Experimental evaluation

First we evaluate the effect of the obtained rate bound on the example of the worst case scenario with three TCP flows from Section 6.1.1. After that in order to complete the picture we test our findings on a larger topology. This time we have six parallel TCP connections and every second

TCP segment size, B	MNP ratio, %	MMF index, f_1	TCP throughput, kb/s Min	Max	Total
200	0	0.97	37	55	148.053
400	0	0.97	60	84	230.151
600	0	0.97	73	104	281.920
800	0	0.97	78	118	314.595

Table 8: The effect of source throttling in 3 TCP flow case.

Original behavior:

TCP segment size, B	MNP ratio, %	MMF index, f_1	TCP throughput, kb/s Min	Max	Total
200	65	0.64	0.8	106	325.546
400	65	0.67	3.0	164	508.053
600	45	0.73	11.8	194	625.866
800	58	0.72	10.3	223	708.906

Applying rate bound at sources:

TCP segment size, B	MNP ratio, %	MMF index, f_1	TCP throughput, kb/s Min	Max	Total
200	0	0.98	37	54	278.400
400	0	0.98	57	84	437.048
600	0	0.98	73	103	541.013
800	0	0.98	84	116	608.497

Table 9: The effect of source throttling in 6 TCP flow case.

connection is placed according to the worst case placement with respect to the lower and upper connections.

Table 8 shows how the situation changed in the three TCP flows case in comparison to the original performance in the worst case (see Table 3). Table 9 summarizes the results from 30 simulation runs of the experiments with six TCP flows for each value of the TCP segment size. The upper table shows the original performance of all TCP flows and the lower table illustrates the effect of limiting the transmission rate at sources according to the derived bound.

In all tables in this section *MNP ratio* stands for maximum no-progress ratio and *MMF index* denotes max-min fairness index as defined in Appendix A equation (5) and the values of these metrics are medians of 30 samples.

As we observe from both experiments applying our bound on sources transmission rates we obtain not only the optimal but almost perfect fairness amongst TCP flows both in three and six flows cases. Recall that the total interference load produced by a connection is an averaged value. Therefore even though applying source throttling the total load from two TCP flows on average is the same as in the case of a single connection the resulting traffic does not contain bursts. Thus less collisions occur between packets of competing flows, which results in better fairness figures.

The 97% - 98% fairness observed in all measurements indicates the accuracy of the bound estimation. If the bound would be overestimated than the resulting source rates would be lower. In this case we would observe the perfect fairness. On the other hand underestimation would result in lower values of fairness. Thus we can conclude that our rate bound (3) is accurate.

Another important observation we make on the resulting individual and total TCP throughput. From the experiments with six flows we see the change dynamics of these characteristics. First of all although the maximum throughput for a flow decreased on average on 80 kb/s when shaping the traffic according to our bound the minimum throughput raised significantly: more than 40 times for 200 bytes TCP segments, 19 times for 400 bytes segments, six times and eight times for 600 and 800 bytes segments respectively.

As for the total TCP throughput in the network we observe an insignificant decrease of it when sources rate is shaped according to our bound. However we argue that the average decrease of this metric on 100 kb/s is a fair price for achieving perfect bandwidth share between all flows.

7 Discussion

The first issue we would like to discuss in this section is why the obtained bound is the upper bound. We built all our reasoning by considering the worst case placement of nodes of the competing connections under the assumption that no one node of one connection is located in the MAC region of another. Considering other placements of nodes which do not contradict the major assumption will overestimate the bound. As we discuss in the following subsection the bound in the case where node placement contradicts the assumption would be strictly less than the obtained one.

7.1 Connections that share their MAC regions

The case where several connections partly or completely share their MAC regions falls under the competence of already existing distributed QoS algorithms.

It is worth noting here that our bound places an additional constraint on the total rate of such flows. This is because in this case we have to consider an influence of the group of flows sharing their MAC regions on the sample connection. It is possible to find such a placement of these connections that all together they will form the area of maximum interference where we can place another connection which satisfies our major assumption now with respect to the group of the connections. Obviously the maximum interference load in the epicenter of AMIs from two groups of such connections is determined by transmissions from all nodes of the involved connections. Therefore in order to prevent TCP capture in this case the total rate of these connections should be less or equal to the obtained bound. This implies that the rate of each source in the group of connections will be strictly less than the obtained bound.

7.2 Flows with shorter routes

For the derivation of the rate bound we considered the case where all involved TCP connections have the path length as a common factor. In the case where interfering flows have shorter routes they have an additional advantage with respect to our sample flow. Obviously for shorter connections more TCP data can be delivered to the destination within same time interval. Thus the interference load created by more than one such connections will be higher. In order to achieve

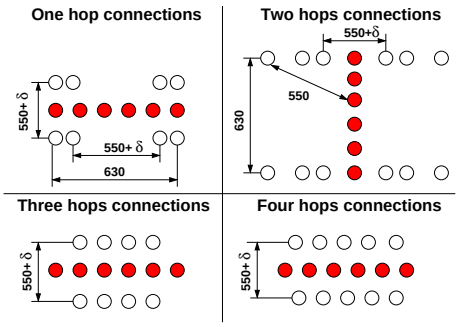


Figure 11: Worst case node placements for connections with shorter routes.

Number of hops	k	n
1	4	1
2	4	2
3	2	3
4	2	4

Table 10: Parameters k and n for computation of the rate bound for connections with shorter routes.

fair capacity distribution in this case sources of shorter connections should limit their transmission rate differently.

It can be shown that the nodes placement shown in Figure 11 is the worst case placement for connections with shorter routes with respect to the sample five hops connection. In the figure the dark circles show the nodes of the affected TCP connection and empty circles show the nodes of interfering connections. The proof of the last claim follows the guideline described in Section 5. Due to its size the complete proof is omitted in this paper and left to the reader. The placement of nodes depicted in the figure dictates the settings of parameters k and n in (3) as shown in Table 10.

7.3 Practical application of the bound

The key step of our methodology for computation of the upper bound for the transmission rate at TCP sources is the construction of the worst case scenario. For this purpose we used radio transmission ranges derived from the NS-2 radio transmission model for an open space environment. In real environments, in particular in indoor or urban environments, the transmission ranges are obviously less and have a complex geometry due to signal attenuation in the construction elements, diffraction and reflection. Nevertheless interferences from other stations communicating over the same radio channel exist in these environments as well and it is essential to account for their effect in the design of wireless networks.

In WLANs [18] such design considerations assume preliminary site investigations based on signal strength measurements and identification of the areas where interferences from other stations transmitting on the same radio channel would cause the decrease in network performance.

After that based on the results of such analysis the network administrators are advised to assign different communication channels to the base stations to prevent the bad effect of interferences. In ad-hoc networks the later step is impossible in general due to the spontaneous nature of these networks. Therefore it is valid to assume that all stations operating in the ad-hoc mode will communicate over the same radio channel, thus the situation described in this paper can appear.

We conjecture that in order to apply our methodology in real environment preliminary estimations of the communication ranges should be made for a particular site. If such estimations show that the geometry and physical properties of the site allow potential formation of the worst case scenario the source throttling according to our bound can be applied. Otherwise in the cases where the interferences do not expose a danger for the multi-hop communications the rate throttling should be disabled since it leads to unnecessary decrease of the total and individual throughput.

As for the case where the source throttling is potentially needed the following implementation choice might be considered. Assume a reactive ad-hoc routing protocol is used to provide connectivity in multi-hop wireless network. Assume as well the routing protocol is enriched with a functionality of gathering information about ongoing connections in the neighborhood of one hop of every node of the establishing connection. The value of one hop is chosen since under the minimum distance placement of nodes the nodes beyond this distance are located outside the communication range of this node. The sources of TCP connections have to limit their rate according to the bound (3) when during the route establishment phase the routing protocol reports absence of competing connections along the path. The parameter n in (3) is easily obtainable from the routing protocol which has information about the path length of the established connection.

7.4 Negative experience of using reactive approaches to solve the TCP capture problem.

In fact the work presented in this paper appeared as a result of several unsuccessful attempts to deploy alternative to distributed algorithms methods to solve the TCP capture problem. In the context of the paper it is worth to mention the two main mechanisms which were under consideration: Application of the end-to-end forward error correction (FEC) schemes and an active disturbance of communications of the winning flows. If the idea of first mechanism is well known the second approach concerns localization of the congestion region ("epicenter" in our terminology) and active disturbance of communications ongoing in the interference range of the affected nodes. The idea behind the second mechanism is simple: If the two flows do not feel the congestion which they created to the affected flow, the nodes of this flow might send the noise traffic prior to the retransmission of a TCP segment in order to "clear" the way for its propagation.

However if we look at values of the maximum interference load in the epicenter of AMIs from two connection (see Table 2) we immediately see that it might correspond to the transmission rate of 3.3 Mb/s. This is more than 1.5 times higher than the transmission capacity of the original IEEE 802.11 standard and nodes in this region are simply not able to surely receive or transmit a packet. Obviously any measures directed on breaking this wall will be unsuccessful. Therefore we conjecture that while mechanisms similar to those described above might improve the performance of TCP in lightly loaded multi-hop wireless networks they are not functional in the worst case.

7.5 Towards creation of a formal model

Obviously experimentally derived bounds using the methodology described in this paper are not tight. Formal analysis of the presented properties is needed. Since the work on creation of such model is outside the scope of the paper here we present a brief outline of a possible way of its creation.

The major term which defines the rate bound formula (3) is the interference load created by transmission of TCP data segments. In this paper we estimated this value experimentally. This is not anything but the throughput of a single TCP flow over a multi-hop wireless network multiplied by the number of hops traversed by the flow. This is because everything what has reached the receiver was transmitted by the intermediate nodes as well.

We learned that the interference load, hence TCP throughput, depends on the size of transmitted segments, under-laying protocols' overhead, internode distance and the number of hops traversed by the flow. Naturally the throughput depends as well on the configuration of the TCP congestion window. Based on all our observations we conjecture that the model should have the following form.

$$Thr = f(MSS, CWIN, C, P_o, d_{i,i+1}, h). \tag{4}$$

In (4) MSS is the maximum segment size, CWIN is the size of the TCP congestion window, C is the transmission capacity of the IEEE 802.11 devices, P_o is the overhead of the lower layers protocols, $d_{i,i+1}$ is the internode distance and finally h is the number of hops of the connection.

The analysis of the related work showed that the number of works aiming at formal modeling of a single TCP flow over multi-hop wireless is very limited. One work of interest presented in [17]. There the authors present a formal model of a TCP connection in a multi-hop wireless network. The work describes an approach to model TCP transmissions over IEEE 802.11 network with enabled RTS/CTS exchange as an embedded Markov chain. The comparison of the model with simulation results shows high accuracy of the model. However as we showed in the paper the throughput of TCP depends also on the internode distances, since on smaller distances more nodes of one connection falls in the interference zone of other nodes of the same connection. We foresee that with certain extensions to this model derivation of (4) is feasible.

8 Conclusions

In this paper we presented a methodology to derive bounds on TCP transmission rates at sources which allow guaranteed capture free communications in the case where all flows are separated by distances larger than the range of assured reception. In this case the problem of TCP capture cannot be solved by any protocol based method. We argued that the only way to achieve fair bandwidth distribution for TCP connection is to statically define the transmission rates at sources of TCP connections taking into account the worst case placement of connections with respect to each other.

The main result of this work is that by limiting the transmission rate at sources of TCP flows according to the derived bound TCP capture does not occur and the fairness close to perfect is achieved for all involved flows.

The bounds presented in this paper are valid for the original IEEE 802.11 standard with the highest available data rate of 2 Mb/s. The methodology for computation of the bounds is naturally

applicable for other modifications of the IEEE 802.11 which define availability of multiple transmission rates on the physical layer. We foresee that the rate bounds in this case will naturally rise, however we leave studies of these standards for our future work.

Acknowledgments

The author would like to thank the following people and organizations for their invaluable comments and assistance which allowed this work be done.

Laboratory of Computer Networks and Architectures (CNA) in Swedish Institute of Computer Science (SICS). Stockholm, Sweden. Dr. Thiemo Voigt, Lic.Tech Ian March.

Networking group of Computer Science Department in University of Basel. Basel Switzerland. Prof. Dr. Christian Tschudin, Dr. Christophe Jelger, Dr. Lidia Yamamoto.

References

[1] K.Tang and M. Gerla, "Fair sharing of MAC under TCP in wireless ad hoc networks" , In *Proc. IEEE MMT'99*, Venice, Italy, Oct 1999.

[2] N.H. Vaidya, P. Bahl and S. Gupta, "Distributed fair scheduling in a wireless LAN", In *Proc. MobiCom'00*,Boston, MA, USA 1999.

[3] 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, MD, USA, 2003.

[4] L. Yang, W.K.G. Seah, and Q. Yin, "Improving fairness among TCP flows crossing wireless and wired networks", In *Proc. of the 4th ACM international symposium on Mobile ad hoc networking and computing*, Annapolis, MD, USA, 2003.

[5] P. Bahl, R. Chandra and J. Dunagan, "SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks", [Online]. Available: http://research.microsoft.com/users/bahl/Papers/Pdf/SSCH.pdf

[6] 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.

[7] R. Ludwig and R. H. Katz, "The Eifel algorithm: making TCP robust against spurious retransmissions", *ACM Comp. Commun. Rev.*, vol. 30, no. 1, Jan. 2000, pp. 30-36.

[8] S. Bhandarkar, N. Sadry, A.L.N. Reddy, N. Vaidya, "TCP-DCR: A novel protocol for tolerating wireless channel errors", *IEEE Transactions on Mobile Computing*, Feb. 2004.

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

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

[11] K. Xu, S. Bae, S. Lee and M. Gerla, "TCP behavior across multihop wireless networks and the wired Internet" , In *Proc. WoWMoM'02*, Atlanta, GA, USA, Sep. 2002.

[12] M. Johansson and L. Xiao, "Cross-layer optimization of wireless networks using nonlinear column generation", In *IEEE Transactions on Wireless Communications*, 2004.

[13] V. Kawadia, Protocols and Architectures for Wireless Adhoc Networks, Ph.D. Thesis, [Online]. Available: http://black.cs l.uiuc.edu/~prkumar/ps_files/04_07_kawadia_thesis.pdf.

[14] Ch.Tschudin and E. Osipov, "Estimating the Ad Hoc Horizon for TCP over IEEE 802.11 Networks", In *Proc.MedHoc'04*, Bodrum, Turkey, June 2004.

[15] Network Simulator NS-2, [Online]. Available: http://www.isi.edu/nsnam/ns/.

[16] Ch. Tschudin, "Lightweight underlay network ad hoc routing (LUNAR) protocol", IETF draft (work in progress), 2004. [Online]. Available: http://www.ietf.org/internet-drafts/draft-tschudin-manet-lunar-00.txt

[17] A.A. Kherani and R. Shorey, "Throughput analysis of TCP in multi-hop wireless networks with IEEE 802.11 MAC", In *Proc. IEEE WCNC'04*, Atlanta, USA, Mar. 2004.

[18] Wireless LAN deployment considerations, Intel website. [Online]. Available: http://www.intel.com

Appendices

A Minimizing the diameter of the connection

In order to find a connection with minimum diameter we build the topology as follows. Firstly we place the end nodes of the connection as close to each other as possible but so that their MAC zones do not intersect (the distance between them is $250 + \delta$ m). Then we place other nodes of the connection but the one before the destination on the circle of radius $r_{MACzone} + \delta$ with the center located in the position of the destination node. When placing these nodes we follow the condition of minimum internode distance (1). Finally we place the last node before the destination so that it satisfies the minimum distance condition with other nodes and is located within the MAC zone of the destination. By following always the minimum internode distance condition and the fact the the source and the destination nodes are located on a minimum linear distance from each other we achieve the minimum possible placement of all nodes (hence minimum diameter of the connection) for a five hop topology. The resulting topology and its graphical solution with respect to parameters *W* and *h* of the AMI is shown in Figure 12. In the figure smaller dashed circles show the MAC zones of the corresponding nodes and two large dotted circles show the βMAC zones of the end nodes on the diameter line.

Let us analyze the constructed topology. In order to satisfy the minimum internode distance condition the linear distance between nodes $(n1, n3)$, $(n2, n4)$, $(n3, n5)$ should be $(250 + \delta)$ m. This condition is satisfied when angle α=30° for the segments between node $n6$ and $(n1, n2)$, $(n2, n3)$, $(n3, n4)$ and $(n4, n5)$. In this case the linear distance $x1$ between $(n1, n2)$, $(n2, n3)$, $(n3, n4)$ is 129.9 m and between $(n4, n5)$ is 129.61 m. Knowing $x1$ and α it is easy to calculate

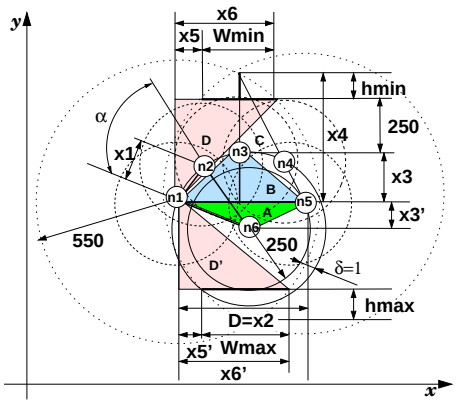


Figure 12: Width and height of AMI for the connection with minimum diameter.

from triangle A the diameter of this connection $D = x2 = 433.9$ m (compare to 630 m for the minimum distance string topology). As we already mentioned the height of the AMI is limited by the MAC zones of most upper and lower nodes of a connection with respect to the diameter line. With the knowledge of the diameter solving triangle A and B we can calculate the height of the corresponding lowest and highest nodes of the connection ($n6$ and $n3$), in our topology $x3' = 125.23$ m and $x3 = 125.73$ m.

The height of the AMI *h* is calculated as $h = x4 - 250 - x3$, where $x4$ is computable solving triangle C. In our topology $h_{min} = 129.673$ m and $h_{max} = 130.173$ m. Finally the width of AMI *W* is computed as $W = x2 - 2 \cdot x5$, where $x5 = x2 - x6$ and $x6$ ($x6'$) is derivable from the solution of triangle D (D′). In our topology $W_{max} = 369.41$ m and $W_{min} = 361.005$ m. This implies that we can place maximum three nodes to form a two hops connection in the AMI of this connection exactly as in the case of the minimum distance string topology. Moreover observe that that even though the diameter of this connection is smaller than in the case of the string topology the height of the AMI is actually decreased. Therefore for further analysis we will use only minimum distance string connections.

A Performance metrics

For the analysis of the fairness properties of more than two TCP flows we use a traditional max-min fairness index (5).

$$f_1 = \frac{(\sum_{i=1}^{n} x_i)^2}{n \sum_{i=1}^{n} x_i^2}. \tag{5}$$

In (5) x_i is the total number of TCP segments delivered by flow TCP_i and n is the total number of TCP flows.

For the convenience of presentation we will also use a simplified version of the fairness index for the analysis of only two TCP flows as defined in (6). This is the ratio between total number of TCP segments delivered to the destination by TCP1 and TCP2. The value of $f_2 > 1$ indicates that TCP1 delivers more traffic than TCP2, when $f_2 < 1$ indicates the opposite situation and the value

of one indicates perfect fairness between the two flows.

$$f_2 = \frac{x_{TCP1}}{x_{TCP2}}. \tag{6}$$

In addition to traditional max-min fairness index (5) we will measure "worst TCP no-progress ratio" metric as defined below. The reason for this is an observation on the performance of the conventional metrics in the analysis of TCP in [14]. The authors show that while fairness index might look fine the TCP connection may suffer from long no-progress intervals.

In order to measure "worst TCP no-progress ratio" 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 some amount of user tolerance, we only count no-progress intervals longer than 1 second. These larger than 1 second intervals are accumulated and put into relation to the total duration of the test run. The higher this ratio is, the more stammering was the session: A 100% *no-progress ratio* means a complete stall.