Drive-thru Internet: IEEE 802.11b for "Automobile" Users

Jörg Ott
Dirk Kutscher

Technologiezentrum Informatik (TZI), Universität Bremen,
Postfach 330440, 28334 Bremen, Germany
Email: {jo|dku}@tzi.uni-bremen.de

***Abstract*— This paper reports on measurement results for the use of IEEE 802.11 networks in *drive-thru scenarios*: we have measured transmission characteristics for sending and receiving high data volumes using UDP and TCP in vehicles moving at different speeds that pass one or more IEEE 802.11 access points at the roadside. We discuss possibilities and limitations for the use of scattered WLAN cells by devices in fast moving vehicles and provide an analysis of the performance that can be expected for the communication in such scenarios. Based on these observations, we discuss implications for higher-layer protocols and applications.**

I. Introduction

Ubiquitous network connectivity – anywhere, anytime – for mobile users and permanent access to the Internet as well as to corporate or private networks has driven many developments in the communications industry over the past years, manifested in the rapid growth and evolution of cellular wide-area networks (such as GSM, GPRS, and UMTS) and wireless local area networks (WLANs, 802.11b/a/g). While cellular networks aim at full coverage (at least in most parts of a country), WLANs are to provide selected hot spots of connectivity, sometimes also covering larger areas such as enterprise premises or university campuses.

Because they strive for full – permanent – connectivity, cellular networks may be referred to as *perma-nets*, WLAN hot spots providing only occasional islands of connectivity as *nearly-nets* [1]. The basic motivation is the same for either type of network: access to public and private online resources; the capability to communicate with co-workers, friends, and family; and, increasingly, to enjoy online entertainment. But arrangements, cost, and services differ: Permanets are expensive to build and maintain, the user base consists of paying subscribers, cost is significant (and thus accounting is important), and the service in terms of affordable bandwidth available for IP communications is limited to some 10 to about 100 kbit/s (GSM to UMTS).

In contrast, WLANs offer only local coverage but provide shared gross data rates from 10 to 50 Mbit/s and have proven to scale to several hundreds concurrently active users when designed properly (as can e.g. be seen from IETF meetings). They support various usage scenarios, with different authentication and tariff models:

- WLANs are often deployed in closed areas such as enterprises (with their use limited to employees) or in semi-public areas such as convention centers (with access usually limited to the participants of a conference), airline lounges, etc. Closed groups are usually established using shared secrets along with WEP-based encryption and authentication. But for frequently changing user communities, more sound security technologies – e.g. network layer security and VPN access – may be used to connect from otherwise strictly separated WLANs to the Internet [2].
- Commercial services have also come up for public places, such as WLAN hot spots in cafés (e.g. Starbucks), at fairs (e.g. the CeBIT in Hannover in 2003), or in shopping malls [3]. Services provided are usually paid Internet access for the users' standard applications: VPN access, mail, web, chat and instant messaging, IP telephony, etc. Those may be augmented by local services offered by the respective wireless service provider, e.g. location-based services, personalized information services, among others [3].
- The service providers get increasing competition from free access services: a community has formed that advocates free access to wireless networks around the globe. Volunteers are encouraged to share their spare bandwidth with others and thereby create a reasonably dense grid of usable access points, particularly in urban areas. Marks on walls ("warchalking") indicate the presence of WLANs and their access conditions (e.g. open or closed), WLAN connectivity maps for cities are created by car (termed "wardriving"), and people worldwide can register (their) access points with global databases for WLANs (e.g. http://www.nodedb.com/, http://www.wifimaps.com/) [4].

Irrespective of how they are managed, today, WLAN-based hot spots can provide a cost-effective and powerful wireless Internet access. However, access is largely limited to local geographic coverage and often to indoor environments. The today less widespread outdoor access for mobile users often comes as a side effect from WLANs inside buildings or from a rather limited number of dedicated access points, e.g. in parks [4], on campuses, or co-located with public phones.

0-7803-8356-7/04/$20.00 (C) 2004 IEEE

But extending WLANs to public (outdoor) places has brought up additional service ideas beyond plain network access: city, shopping, and restaurant guides with location-based services, free access to local information in malls (e.g. advertisements), finding nearby "buddies" [3], among others.

But today's deployments of WLAN-based access technology are mostly limited to rather stationary users: indoor users who are mobile but within a rather limited range and outdoor users who are expected to stop by (in a café or a park area) or to move at most slowly to use the WLAN. Only two types of scenarios for mobile users have also been considered: 1) Providing network services in mobile environments (e.g. airplanes, trains) [5], relative to which the user does not move. 2) Investigating ad-hoc networking scenarios in which several mobile users use WLANs to establish cooperation environments – but without providing Internet access. In [6], Esbjörnsson et al. provide a description of an application for WLAN-based ad-hoc communication between vehicles and in [7], Singh et al. provide some initial measurement results for inter-vehicle ad-hoc communication, however, without considering TCP and communication between a mobile host and hosts on the global Internet. The Fleetnet project [8] has also focused on wireless ad-hoc communication between vehicles but has also investigated the use of ad-hoc communication as a means to extend the reach of existing network infrastructures.

In this paper, we investigate the usability of providing network connectivity and, ultimately, Internet access to mobile users in vehicles. The idea of *Drive-thru Internet* is to provide hot spots along the road – within a city, on a highway, or even on high-speed freeways such as autobahns. They need to be placed in a way that a vehicle driving by will obtain WLAN access for some (relatively short) period of time; if located in rest areas, the driver may exit and pass by slowly or even stop to prolong the connectivity period. One or more locally interconnected access points form a so-called *connectivity island* that may provide local services as well as Internet access. Several of these connectivity islands along a road or in the same geographic area may be interconnected and cooperate to provide *network access with intermittent connectivity* for a larger area.

While this type of networking environment also constitutes some kind of nearlynet as traditional hot spots do, it displays significantly different characteristics due to the usually short-lived connectivity periods. We expect implications for communications at all layers: wireless link, network (IP), transport, and application layer. In this paper, we focus on plain WLAN connectivity and transport protocol behavior – and only briefly address implications on applications in the end. Our goal is to prove that WLAN technology is capable of enabling Drive-thru Internet access in the first place and to document the communication characteristics we have observed with different measurement configurations using UDP and TCP as standard transport protocols.

This paper is organized as follows: in section II, we introduce the idea of Drive-thru Internet in more detail. In section III, we present the measurement configurations that we have chosen, and we document the measurement results in section IV. Section V summarizes the findings from selected measurements, evaluating the results and highlighting issues most relevant to Drive-thru Internet access. Based on these observations, section VI, touches on some of the implications for applications and higher layer protocols. Finally, section VII concludes this paper confirming the suitability of WLAN-based access technologies for Drive-thru Internet and pointing out next steps in our research.

II. Introducing Drive-thru Internet

A generic scenario for user mobility with users moving individually at varying speeds is a vehicle moving along a street (within a city, on highways in the countryside, or on the autobahn). [1] Users may move, stop, continue to move, etc. Users may move at varying speeds ranging from a less than one to some 70 meters per second.

Users may have portable devices (laptops, PDAs, etc.) equipped with a WLAN card, or they may be connected to some in-vehicle networking infrastructure which may provide a WLAN interface to the outside as well as an Ethernet jack to connect to and act as a router for other devices inside the vehicle. An external antenna (e.g. integrated or co-located with a cellular antenna) is used for improved WLAN signal strength – but built-in antennae of WLAN cards or laptops may suffice for some applications.

Access points may be provided at each street corner, co-located with traffic lights, or emergency phones (which are placed every 2 km on a German autobahn), be placed in parking lots or in rest areas or may be co-located with gas stations or other shops in service areas. Several access points may be grouped to extend the reach of a connectivity island.

When driving, mobile devices may have free line of sight to the access point(s) on the roadside; or the access point(s) may be obscured by trees, fences, the user's own vehicle's bodywork, other vehicles, crash-barriers, or even buildings. The view on the access point(s) potentially depends on the roadside the vehicle is driving, the lane, the density of other traffic, etc. That is, WLAN connectivity will appear and disappear; short periods of connectivity will alternate with long periods of non-connectivity; and even the short connectivity periods may be interrupted further. Hence, a mobile device in a vehicle traveling along a road with usable access points occasionally located close to the road will:

1) permanently scan for signals from available access points; [2]
2) attempt to associate with the respective access point whenever such a signal is detected;
3) detect network access;
4) perform IP configuration (obtaining an IP address, performing neighbor discovery) for the respective link after

[1] Other variants of this scenario include (but are not limited to) users traveling by train or public transportation (with the respective vehicles not being equipped with their own means for Internet access). While details may differ, the basic technical considerations are largely similar.

[2] Fortunately, power consumption is less an issue when traveling by car.

the association succeeds in order to be able to send and receive data;
5) use the wireless network for general Internet access, for VPN tunneling, etc. using regular Internet protocols after IP connectivity has been established;
6) go through a series of hand-overs if a connectivity island is made up of several access points;
7) at some point, notice a weakened signal and eventually loss of the signal when the vehicle has passed through the connectivity island and travel on returning to step 1.

With the above scenario, we assume a single access point with an estimated reach of some 200 meters in diameter, located close to a road with direct line of sight and an empty street. In such a setting, a user's device in a vehicle will be in range for about 12 seconds when driving in a city, for about half the time on a speed-limited highway (120 km/h), or for about 3 seconds on a non-speed-limited section of a German autobahn. A sequence of access points along a rest area may prolong this connectivity period. Measurements have shown that it takes about 20-40 seconds to pass by a small rest area (without food/gas) from entry to exit at a speed of some 120km/h.

Starting from such a setting, the focus of this paper is to investigate the actual usability of WLAN in the aforementioned scenario, to validate the rough estimate on the connectivity period given above, and to determine the transmission characteristics observed while passing through a connectivity island. The following sections introduce the measurements we have carried out for Drive-thru Internet and discuss our findings. If our simplified considerations above are confirmed, they indicate that traditional ways of providing Internet access are suboptimal when simply applied to Drive-thru environments: as the connectivity periods are apparently too short for many applications (web, mail, interpersonal communications) and the typical interactive user behavior. We will also briefly discuss some of the potential implications for applications in section VI.

III. Measurement Scenarios

We have performed three different measurement runs: a set of reference measurements in our laboratory, a set of measurements with one access point and a vehicle-borne mobile station at lower speeds (40 km/h to 80 km/h) on a highway, and a set of measurements with two access points and a vehicle-borne mobile station at higher speeds (80 km/h to 180 km/h) on an autobahn (freeway).

For all scenarios, we have used the following components: a mobile station (a laptop with a PCMCIA IEEE 802.11b adapter), a fixed station (a laptop with a built-in Ethernet adapter), and an IEEE 802.11b access point. For the mobile measurements in moving vehicles, we have equipped the PCMCIA card with an external antenna that has been placed at the right hand side of the vehicle and has overtopped the vehicle roof slightly, as depicted in figure 3. [3] The access point and the client adapter were configured to use the same ESSID, WEP encryption has been deactivated, and we have used a beacon interval of one second which is the default settings for most access points. As we were primarily interested in basic transmission characteristics, the mobile device had a statically assigned IP address.

We have measured both UDP and TCP performance in different scenarios. For all UDP measurements, we have used two tools, one for configurable packet transmission (including statistics reporting) and a receiving tool providing detailed logs for the incoming packets. We have transmitted packets of different sizes and in different intervals. In each measurement, we have used one active sender station transmitting packets to the other, using UDP/IPv4 unicast; our tests covered both directions, fixed (Ethernet-based) laptop to mobile and vice versa.

For all TCP measurements, we have used a client and a server, with the client residing on the mobile host, i.e., in the vehicle, and the server running on a fixed Ethernet-based host. Upon entering a WLAN zone, the client connected to the server and initiated a data exchange according to a test specification. Both sides periodically reported the transmitted bytes per time frame until the connection was interrupted due to a loss of the connection from the mobile host to the access point.

A. Measurement Tools

Our UDP-based tools have used RTP [9] as a transport protocol. In particular, we have used RTP sequence numbers to assess throughput and packet loss, and we have used RTP timestamps plus additional (finer-grained) timing information in the actual payload to monitor relative packet delays.

Our sending tool *rtpsend*[4] can be parameterized with an IPv4 destination address and a port number, with an interval between consecutive packets and with an RTP packet size, i.e. the size of the whole UDP packet including the twelve bytes for the RTP header.

The receiving tool *rtpspy* logs each received packets and its RTP header information, especially the sequence number and the RTP timestamp plus, optionally, additional timing information contained in the payload. Based on the complete list of all received packets within a time frame, *rtpspy* can generate statistics for a session, e.g., on the average throughput, packet loss totals and histograms of consecutive packet loss. Using the RTP timestamp, *rtpspy* can calculate *relative* transmission delays by considering all received packets and comparing their RTP and payload timestamps (sending time) with the time of receiving the packet.[5] We have extracted further statistics from the log files by means of extensive scripting.

For most measurements, we have used packet sizes of 1250 bytes and have used sending intervals of 4ms, 2ms, and 1ms, i.e., 250, 500, and 1000 packets per second.

[3]The access point was a Cisco Aironet 340 system, and the PCMCIA adapter was a Orinoco 802.11b "Gold" card.

[4]Our *rtpsend* is not the rtpsend tool from Columbia University.

[5]For *absolute* delay calculations, synchronized clocks between sender and receiver would be required, which we did not rely on for these measurements.

For TCP measurements, we have developed a tool called *tcpx* (TCP exchange). *tcpx* can be started in either server mode or client mode. In server mode, *tcpx* will multicast periodic UDP trigger messages and listen for a new connection on a specified port. In client mode, *tcpx* will wait until it receives a trigger message from the server, e.g., when the client system enters the local network in which the server resides. The trigger mechanism is used to automate the reachability detection and the connection setup when the mobile station enters the Drive-thru WLAN cloud. Upon reception of a trigger message, the *tcpx* client establishes a connection and transmits a TCP message exchange specification to the server, thus defining the communication characteristics of the TCP session. Communication is modeled as a series of request-response interactions, where the client sends a certain number of bytes and the server responds with a certain number of bytes. The request and response message size can be parameterized when starting the client and is sent to the server in the initial configuration message. The client and the server are synchronized, i.e., each party waits until the specified number of bytes has been received before it starts sending.

After the initial configuration message, the client and the server start to exchange bytes following the specified pattern for a user-defined amount of time or until the TCP connection is lost. Both the client and the server periodically report the number of bytes sent and received. The interval between these reports is configurable, and for our tests, we have used 300 ms.

In order to compare TCP behavior to the UDP measurements and in order to measure the throughput in one direction, we have configured the *tcpx* instances to perform a one way communication, i.e., either the client or the server sends.

For the TCP measurements, we also generated network traffic traces using *Ethereal*[6]. These traces have later been processed with *tcptrace*[7].

In addition to RTP and TCP transmission and reception statistics, we have also measured WLAN characteristics. We have used the tool *NetStumbler*[8] on the mobile station and have connected a GPS receiver for logging position information.

B. Reference Scenario

In a reference measurement run in our laboratory, we have tested different configurations under optimal 802.11 conditions, using a wireless station that was immobile and very close to the access point.

C. Highway Scenario

For an initial set of mobile measurements, we have set up a single access point and a fixed station at the roadside and have performed some lower-speed measurements with the mobile system in a vehicle acting as the sender.

Figure 1 depicts the setup for this scenario: the access point has been directly connected to the fixed station. We have performed some measurements with 40/km, 60 km/h and 80 km/h. These measurements have shown that WLAN networking with moving vehicles is generally feasible and have led to the design of the actual autobahn measurement scenario, which we describe in the following section.

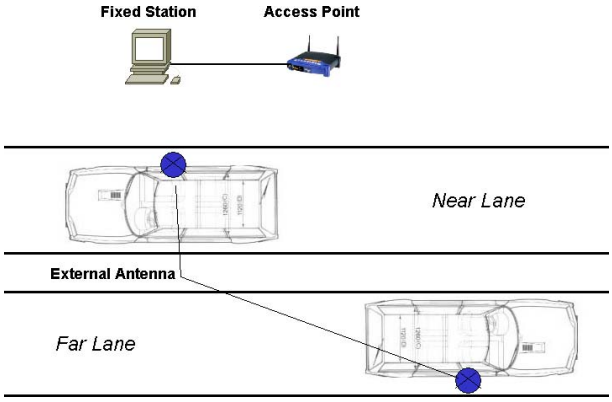

Fig. 1. Highway measurement configuration

D. Autobahn Scenario

The actual measurements have been conducted on a freeway, at both freeway speed (120 km/h to 180 km/h) and at lower speed (80 km/h) in order to approximate highway driving conditions.[9]

The freeway measurements have been performed on an uncongested german four-lane autobahn.[10] We have placed two access points at a rest area, close to the autobahn, in a distance of 100 meters to each other. Figure 2 depicts this setup. The section of the street that was covered by the access points was straight without any bends thus offering optimal visibility. The access points have been configured to use different channels (6 and 11).

We have performed the measurements in both directions, north and south; we did not differentiate between the two lanes that go into each direction. Generally, for lower speeds (80 km/h and 120 km/h) we have held onto the right hand side, and for 180 km/h[11], we have used the left lane. For all measurements, we have controlled the vehicle speed with the GPS receiver and have maintained a steady speed using cruise control.

As depicted by figure 3, the external antenna was mounted at the right hand side of the vehicle. Hence, for driving north, the antenna was on the near side of the access point, whereas for driving south, the antenna was on the far side.

A few measurements on the autobahn *without* an external antenna have revealed that an external antenna is absolutely

[6]http://www.ethereal.com/

[7]http://irg.cs.ohiou.edu/software/tcptrace/tcptrace.html

[8]http://www.stumbler.net/

[9]Initial highway measurements have shown that for the chosen access point configuration, the communication characteristics for lower speeds (50 km/h to 80 km/h) do not differ significantly. In order to obtain comparable results, we have conducted all detailed measurements using the same freeway configuration.

[10]On the A27 between Uthlede and Hagen, northern Germany

[11]80 km/h is approximately 50 mp/h, 120 km/h is approximately 75 mp/h, and 180 km/h is approximately 112 mp/h (1 kilometer = 0.6213712 miles).

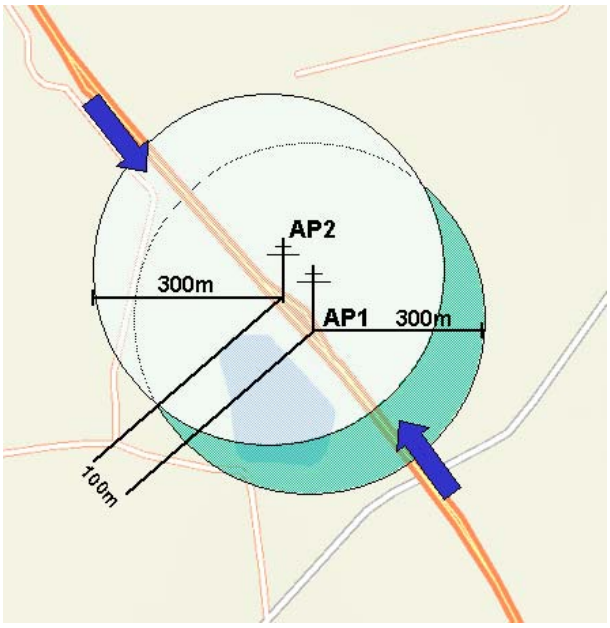

Fig. 2. Autobahn measurement scenario

required. Although we have been able to receive a few beacons during an antenna-less test drive, we have not been able to either send or receive a single packet.[12]

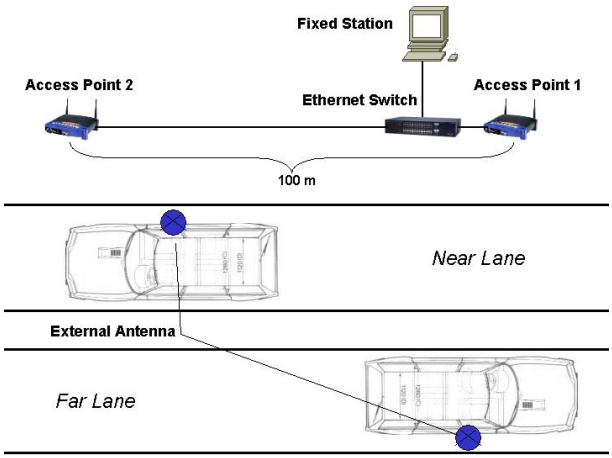

Fig. 3. Autobahn measurement configuration

IV. Measurements

In the following, we first report on results from our reference measurements in the laboratory (section IV-A), from which we have derived interesting measurement configurations for the final autobahn measurements. In general, we have first investigated the communication characteristics using unreliable UDP transport and have subsequently conducted corresponding TCP tests in order to determine TCP's performance under the observed conditions. The UDP measurement results are described in section IV-B, and the TCP measurement results are described in IV-C.

[12]Tests with an external antenna *inside* the vehicle still need to be carried out.

A. Reference Measurements

The objective for the reference measurements was to determine the maximum throughput under optimal conditions in order to derive useful settings for the actual autobahn measurements. Table I depicts the different configurations for our UDP measurements. The TCP reference measurements are described at the end of this subsection. For each measurement, the *Sender* column specifies the sender (*mobile* designates the WLAN station and *fixed* designates the station on the fixed Ethernet).

The *nominal sending data rate* is derived from the parameters *packet size* and *interval* – it is not the effective sending rate (as the results will reveal). Obviously, measurement runs #3 and #7 exhibit sending rates that are well below the achievable IEEE 802.11b net throughput. We have performed these measurements in order to determine the maximum throughput under optimal conditions.

ID	Sender	Packet size	Interval	Nominal sending rate
1	fixed	1250 bytes	4 ms	2.5 Mbit/s
2	fixed	1250 bytes	2 ms	5 Mbit/s
3	fixed	1250 bytes	1 ms	10 Mbit/s
4	fixed	125 bytes	2 ms	0.5 Mbit/s
5	fixed	125 bytes	1 ms	1 Mbit/s
6	mobile	1250 bytes	2 ms	5 Mbit/s
7	mobile	1250 bytes	1 ms	10 Mbit/s
8	mobile	125 bytes	2 ms	0.5 Mbit/s
9	mobile	125 bytes	1 ms	1 Mbit/s

TABLE I
Reference UDP measurements

We have chosen 1250 and 125 bytes as packet sizes because we wanted to contrast smaller and larger packets, however without causing fragmentation. We expect the maximum throughput to be approximately 5 Mbit/s, which can be achieved by sending 1250 bytes packets with a 2 ms interval (500 packets per second). Therefore, we have measured the behavior with 1 ms, 2 ms, and 4 ms intervals.

Table II depicts some UDP measurement results. In order to facilitate the comparison between nominal sending rate and receive rate, we have included the nominal sending rate again. The *effective throughput* is the throughput as observed by the *rtpspy* tool, i.e., the receiver. Accordingly, the *loss rate* is the packet loss rate as observed by *rtpspy*.

The most obvious result is the different behavior with respect to the role of the sender: for a mobile sender (measurements #6 through #9), we experienced almost no packet loss, whereas for a fixed sender, the loss rate is depending on the nominal sending rate and can be as high as 62.1%, when the network is heavily overloaded. The results from measurement #7 show that, although there is almost no packet loss, the nominal throughput of 10 Mbit/s cannot be achieved at all. Instead the effective throughput is about 5 Mbit/s, which

ID	Nominal sending rate	Effective throughput	Loss rate
1	2.5 Mbit/s	2.38 Mbit/s	4.81%
2	5 Mbit/s	3.38 Mbit/s	31.09%
3	10 Mbit/s	3.79 Mbit/s	62.10%
4	0.5 Mbit/s	0.47 Mbit/s	4.48%
5	1 Mbit/s	0.56 Mbit/s	43.72%
6	5 Mbit/s	4.92 Mbit/s	0.00%
7	10 Mbit/s	5.04 Mbit/s	0.01%
8	0.5 Mbit/s	0.49 Mbit/s	0.00%
9	1 Mbit/s	0.95 Mbit/s	0.01%

TABLE II
REFERENCE UDP MEASUREMENT RESULTS

means, as there is almost no packet loss, that the effective *sending rate* is also at about 5 Mbit/s.

This deviation of the effective sending rate from the nominal sending rate for mobile senders, i.e., stations with an IEEE 802.11b network interface, is caused by the IEEE 802.11b media access mechanism [10] – the *Distributed Coordination Function* (DCF). DCF relies on a CSMA/CA approach and defines an algorithm by which a sender tries to allocate sending slots in order to avoid collisions. The allocation of sending slots can be delayed, e.g., when other stations are sending or when the sending rate is too high.

In general, some IEEE 802.11 implementations (such as the ORiNOCO adapter that has been used here) tend to *block* the sending request from the upper layer in these cases (under the assumption that sending slots will be allocated in the near future and that the operation can complete). This results in an *implicit flow control* that is imposed on the sender by the IEEE 802.11 MAC layer: the sending application is throttled down to the maximum throughput that can be sent using the DCF medium access mechanism.

For *fixed* senders, this implicit flow control does not apply, because the fixed stations do not provide an IEEE 802.11b interface on their own. Instead, they send packets via their Ethernet interface to the access point that performs the wireless transmission. Depending on the queue buffer size of the access point, it can also delay packets for short periods of time, however, there is no feedback loop to the sender. In the presence of sustained network congestion, the access point will eventually have to drop packets, which leads to the significantly higher loss rates when sending with higher sending rates from a fixed sender that is not directly connected to the wireless network.[13]

With respect to *transmission delay*, we have observed a similar behavior as overloaded routers exhibit when queue buffers fill up: the transmission delay increases until the maximum queue size is reached, and subsequently, packets will be dropped (drop-tail mechanism), which becomes visible by an increased packet loss rate.

With these considerations in mind, we can conclude that the maximum throughput that can be achieved when sending from a mobile to a fixed station is approximately 5 Mbit/s, which we have achieved by measurement #6, where we have sent 1250 byte packets at a 2ms interval. When we double the nominal sending rate as done in measurement #7, the sender can still not achieve a higher throughput but is also throttled down to 5 Mbit/s.

Interestingly, when using the sending rate of measurement #6 in the opposite direction, i.e., from fixed to mobile (measurement #2), we experience severe packet loss (31.09%) and achieve a net throughput of only 3.38 Mbit/s. Even with half the nominal sending rate (2.5 Mbit/s in measurement #1), we still experience approximately 5% packet loss.

We can also note that, when sending from fixed to mobile with a high packet rate, e.g., 1000 packets per seconds as in measurement #5, we also observe extraordinary high packet loss rates, although the sending data rate is only 1 Mbit/s. In the opposite direction, however, we have managed to send 1 Mbit/s with 1000 packets per second.

Taking these observations for our reference measurements into account, we can limit the interesting measurement cases for the autobahn scenario as follows:

- for sending from mobile to fixed, we can limit the sending rate to a nominal rate of 5 Mbit/s;
- for sending from fixed to mobile, 2.5 Mbit/s can already reach the limit of the network's capacity; and
- for sending from fixed to mobile, we can expect very high loss rates for higher sending rates.

For our TCP reference measurements, we have determined the maximum throughput for bulk data transmission in both directions, i.e., mobile to fixed and fixed to mobile. For sending from mobile to fixed, the maximum TCP throughput was 4.38 MBit/s, and for sending from fixed to mobile, the maximum TCP throughput was 4.44 MBit/s. These figures refer to the average throughput of payload data (without the TCP header).

This means, for sending from mobile to fixed, we have not quite achieved the maximum UDP throughput of approximately 5 MBit/s, whereas for sending from fixed to mobile, TCP performs significantly better than UDP (3.79 MBit/s). We ascribe this to TCP's congestion control that results in a more efficient use of the available bandwidth compared to UDP.

B. Autobahn UDP Measurements

With the observations from our reference measurements in mind, we have defined seven UDP measurement configurations for the autobahn scenario. Table III provides an overview of the different measurement configurations. We have performed each measurement twice: once on the *near side* (driving north) and once on the *far side* (driving south).

Table IV depicts the corresponding overall throughput per seconds and the loss rate. For each measurement, this refers to the time from the first to the last packet that has been *received*.

Similar to our reference measurements, these result show significantly different loss rates depending on the role of the sender: for mobile senders, we observe loss rates with a maximum of 1.43% (measurement #6), whereas for the opposite direction, we have observed a *minimum* loss rate of 57.18%

[13] It should be noted that mobile senders will exhibit a similar behavior as soon as they are connected to a local router, e.g. as part of the vehicle infrastructure.

ID	Speed	Sender	Packet size	Interval	Nominal sending rate
1	120	mobile	1250 bytes	2 ms	5 Mbit/s
2	80	fixed	1250 bytes	4 ms	2.5 Mbit/s
3	120	fixed	1250 bytes	4 ms	2.5 Mbit/s
4	180	fixed	1250 bytes	4 ms	2.5 Mbit/s
5	80	mobile	1250 bytes	2 ms	5 Mbit/s
6	180	mobile	1250 bytes	2 ms	5 Mbit/s
7	120	fixed	125 bytes	1 ms	1 Mbit/s

TABLE III
AUTOBAHN MEASUREMENTS

(measurement #2). Again, this can be explained by the IEEE 802.11 medium access mechanism. In our measurements, the fixed sender has been sending continuously, and the first packet has been delivered when the mobile station initially enters the range of the access points. Due the variability in signal quality, especially for long distances, the access point fails to send all packets in time, builds up a queue and finally has to drop packets, while the sender continues to send at its nominal sending rate.

ID	Nominal sending rate	Effective throughput	loss rate
1 (near)	5 Mbit/s	0.74 Mbit/s	0.52%
1 (far)	5 Mbit/s	0.88 Mbit/s	0.82%
2 (near)	2.5 Mbit/s	0.75 Mbit/s	57.18%
2 (far)	2.5 Mbit/s	0.76 Mbit/s	43.65%
3 (near)	2.5 Mbit/s	0.42 Mbit/s	63.36%
3 (far)	2.5 Mbit/s	0.62 Mbit/s	52.43%
4 (near)	2.5 Mbit/s	0.43 Mbit/s	62.74%
4 (far)	2.5 Mbit/s	0.17 Mbit/s	57.85%
5 (near)	5 Mbit/s	1.43 Mbit/s	1.02%
5 (far)	5 Mbit/s	1.15 Mbit/s	0.68%
6 (near)	5 Mbit/s	0.82 Mbit/s	0.67%
6 (far)	5 Mbit/s	0.29 Mbit/s	1.43%
7 (near)	1 Mbit/s	0.11 Mbit/s	82.93%
7 (far)	1 Mbit/s	0.05 Mbit/s	84.00%

TABLE IV
AUTOBAHN MEASUREMENT RESULTS

Given the variability of signal quality over distance, it is obvious that we have to consider the temporal distribution of throughput and also have to take delay into account in order to analyze the transmission characteristics. Figure 4 depicts the throughput for the "North" variants of measurements #1, #5 and #6 (5 Mbit/s from a mobile sender with 1250 bytes packets every 2 ms at different speeds). For each graph, the measurements start from the first packet received at the fixed receiver. Given a constant speed, we have mapped the temporal distance to geographical distance in order to compare the results at different speeds. We have not aligned the different graphs, e.g., by shifting the start times.

We can notice the expected variability in throughput: at large distances, the throughput is well below 1 Mbit/s, whereas for all speeds, there is a range of approximately 250 meters where throughput reaches approximately 4 Mbit/s – almost irrespective of the speed.

When we accumulate all packets of these measurements, we can determine the absolute data volume that we can transmit

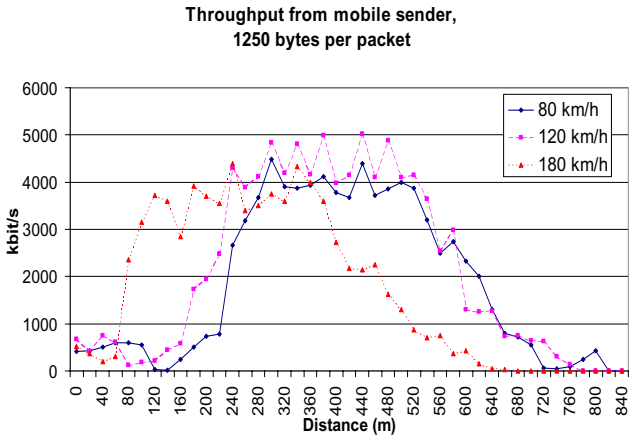

Fig. 4. Throughput from a mobile sender at different speeds

in one Drive-thru session. Figure 5 depicts the total for the different speeds over distance, again determined from the temporal distribution starting from the first packet received.

As one could expect, the total data volume is dependent on the vehicle speed. In fact, the accumulated data volume is decreasing proportionally with the vehicle speed. We can determine a total data volume of approximately 8.8 MByte for 80 km/h, 7.8 MByte for 120 km/h, and 3.7 MByte for 180 km/h. Corresponding to figure 4, we can observe a range of about 250 meters, where the data volume is increasing linearly, whereas the gradient is greater for the lower speeds, i.e., the effective throughput per time is slightly higher.

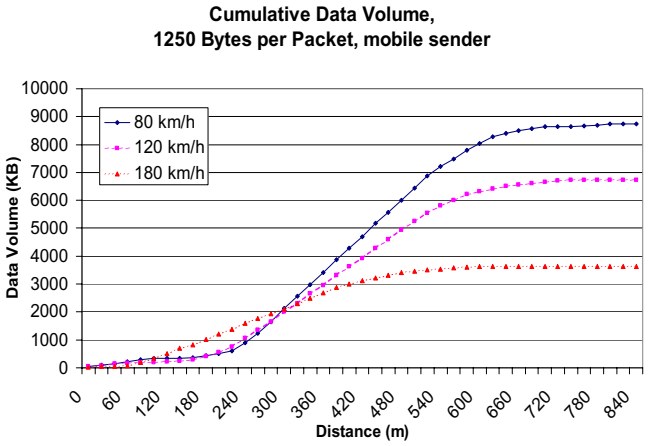

Fig. 5. Cumulative data volume at different speeds (mobile sender)

Figure 6 compares the relative delays between packets for the different speeds, as observed by the fixed receiver. For all speeds, the sender builds up a delay after starting to send, which largely disappears as soon as the throughput increases. After about 500 meters, the delay per packet starts to increase again and reaches a similar level as for the beginning of the drive-thru session.

This initial and final delay can be explained by the degra-

dation of the signal quality, which leads to link layer retransmissions, thus increasing the delay for packets that have been accepted for transmission by the IEEE 802.11 MAC layer. As soon as the signal quality and the throughput improves, the delay is decreasing quickly.

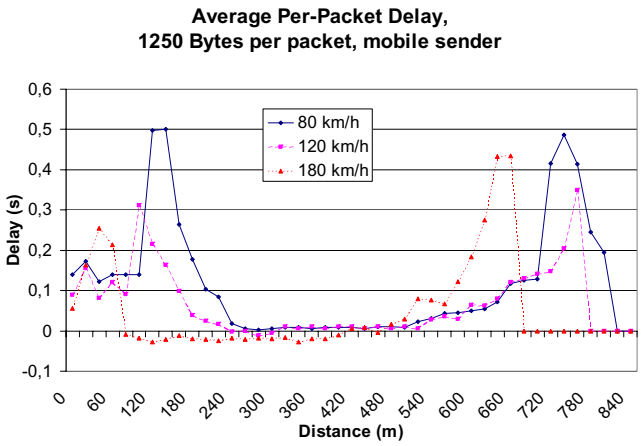

Fig. 6. Relative per-packet delay at different speeds (mobile sender)

When we compare the cumulative throughput from the mobile-sender scenario to the fixed sender scenario, we can observe significant differences. Figure 7 depicts the accumulation of received data that has been sent by the *fixed* sender. Except for the lowest speed (80 km/h), the total number of bytes does not even reach 50% of the figures for the mobile sender. The difference between throughput per time for different speeds is also more significant as it was the case for the mobile sender scenario.

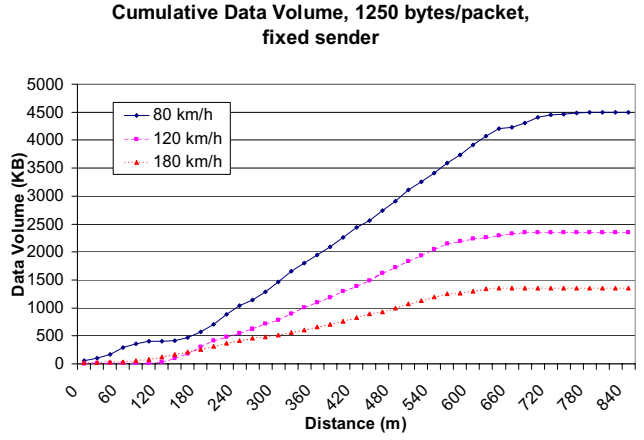

Fig. 7. Cumulative UDP data volume at different speeds (fixed sender)

C. Autobahn TCP Measurements

The results of the UDP measurements indicate that the communication characteristics (packet loss rate, delays) change rapidly in a mobile scenario. We have noted a significant variability in throughput over time – however even at higher speeds an accumulated throughput per Drive-thru cloud of at least 3.8 MBytes has been achieved. In this section, we present some measurement results for TCP communication in order to investigate how TCP can perform under these conditions.

Similar to our UDP measurements, we have performed TCP measurements at three different speeds: 80 km/h, 120 km/h, and 180 km/h. For all speeds, we have transmitted bulk data in different tests for each direction (mobile to fixed and fixed to mobile). One interesting observation was that for TCP, we did not note a distinct asymmetric behavior for sending from mobile to fixed and vice versa. For UDP, sending from mobile to fixed exhibited better performance than sending from fixed to mobile, whereas for TCP, we have achieved similar throughput rates for both directions. In the following, we will present some results for sending from fixed to mobile that are representative for the corresponding measurements for sending from mobile to fixed.

Figure 8 compares the throughput for different speeds at different distances from a virtual starting point when sending from a fixed sender to a mobile receiver. We can observe a temporary maximum throughput of almost 4.5 MBit/s (for 120 km/h), however with a significant amount of variability. At 180 km/h, we experience an even higher degree of variability and a lower average throughput. To some extent the variability can be ascribed to the resolution of the measurements, but in any case it is notable, that for 180 km/h the maximum throughput is lower than for 120 km/h and 80 km/h.

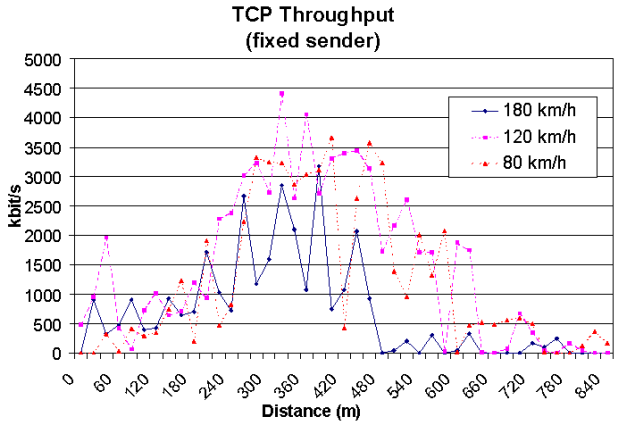

Fig. 8. TCP throughput (fixed sender)

Figure 9 depicts the cumulative TCP data volume that has been transmitted at different speeds. For 120 km/h, we achieve a total throughput of almost 5 MBytes, and for 80 km/h, we have managed to transmit up to 6 MBytes in a single TCP session. At 180 km/h, the overall throughput is significantly lower (1.5 MBytes). It should be noted that these numbers are significantly higher than the corresponding numbers for transmitting from fixed to mobile with UDP as depicted in figure 7. Especially for 120 km/h, the TCP is able to increase

the overall throughput by a factor of 2. However, we can note that the cumulative data volume for a TCP session at 180 km/h is superproportionally lower compared to the throughput for lower speeds – different to the results of the corresponding UDP measurements depicted in figure 7. We ascribe this to retransmissions that are caused by occasional transmission failures and to a faster switching from different IEEE 802.11 sending rates to which TCP cannot adopt fast enough.

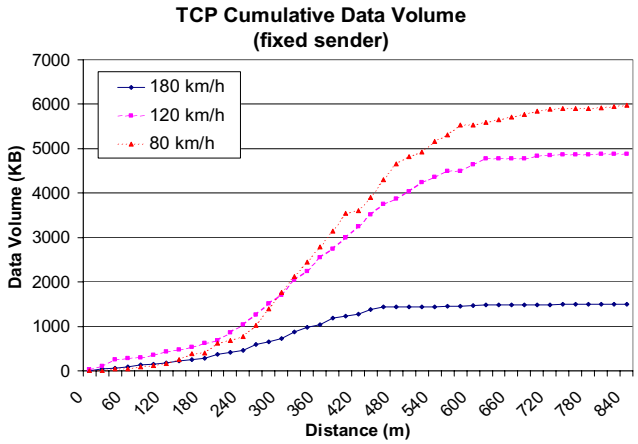

Fig. 9. Cumulative TCP data volume at different speeds (fixed sender)

For analyzing the implications of TCP's retransmission and congestion control mechanisms with respect to the observed results we have monitored the TCP communication and have generated a time sequence graph in figure 10 that depicts the increase of the sequence number values over time. We can see an initial period of poor connectivity, where sequence number values are only increasing slowly, which is a result of many retransmission (depicted by the character R in the figure). After approximately ten seconds, the reliability of the link stabilizes and we see a continuous increase of sequence numbers without TCP retransmissions. After another 8 seconds, transmission failures lead to an increasing number of TCP retransmissions and the sequence number gradient decrease again.

V. Experimental Findings and Analysis

From the discussion of the measurement results for the autobahn scenario, we can derive the following conclusions.

The first observation is that IEEE 802.11b communication with mobile stations is essentially feasible, irrespective of speed. Even for 180 km/h (faster than average on a German autobahn), we have been able to transmit up to 3.8 MByte for a whole Drive-thru session.

Naturally, there is a certain window of *useful* connectivity – referred to as the *production phase* – available during which effective communication can take place using both UDP and TCP. Initially, during the *entry phase*, the connectivity is weak: packets can already be transmitted, however, the loss rates and the delay measured for UDP are too high to allow for the sustained transmission of many packets. With TCP, we have observed delays of up 2.5 seconds before the SYN/ACK

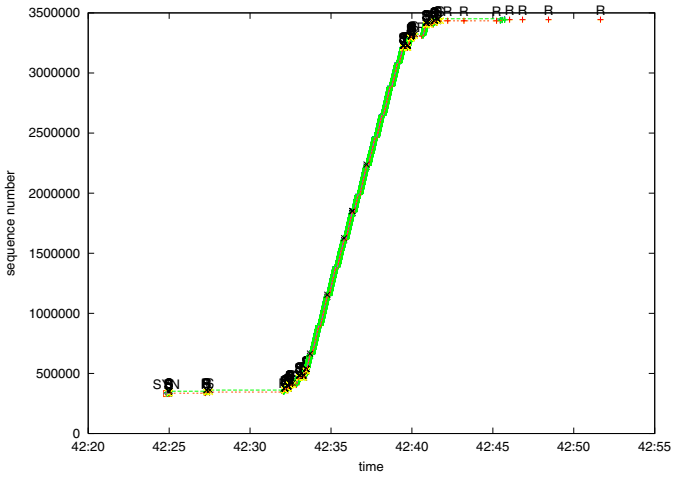

Fig. 10. TCP time sequence graph (mobile to fixed at 120 km/h)

handshake for connection setup completed. When connectivity fades during the *exit phase*, loss rates and delay increase again.

Figure 10 clearly shows these three distinct phases of the communication for a single TCP session, which has also been indicated by our UDP results. Figure 11 depicts these different TCP connectivity phases with respect to the throughput and the distance from a virtual common starting point for measurements at different speeds.[14] The *entry and exit phases* exhibit low throughput rates. The switch to the *production phase* occurs quite quick allowing for significant higher throughput rates.

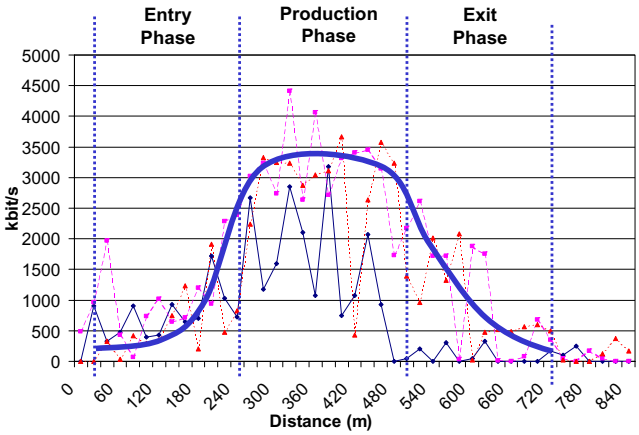

Fig. 11. TCP connectivity phases

From the diagrams in the previous sections, we can estimate that the window of useful connectivity is at least 200m in diameter, resulting in about 9 seconds at 80 km/h, 6 seconds at 120 km/h, and 4 seconds at 180 km/h. This connectivity window is equally usable for both UDP and TCP. If used at

[14] A similar modeling can also be derived from our UDP measurements, e.g. based upon figure 4.

the right transmission rate, UDP flows experience only little packet losses. TCP's congestion control mechanisms make TCP flows adapt quickly to the varying networking conditions and provide reliable communications at decent throughput. It should be noted, however, that TCP's quick adaptation is likely to be related to the wireless LAN being a one-hop network setup without additional delays and losses incurred by access and backbone links. Further investigations are required to study implications of communicating with more remotely located peers on TCP's behavior in a Drive-thru scenario (see also section VI).

Although we have set up the autobahn scenario with two access points to extend the connectivity window, we have observed that there is no handover at all. Irrespective of speed and direction, the client adapter has always associated with the access point geographically closer upon arrival – access point #1 when driving North and access point #2 when driving South – and has never dissociated from this access point. We ascribe this to the scanning rate of the client adapter firmware that is too low to allow for a fast handover in this scenario. Since both access points have been configured to use different channels, the client adapter has to scan multiple channels – at a time where it is busy sending to (or receiving from) from one access point. In addition, the access points were comparably close together, which was due to space restrictions at the rest area.

This means, we have achieved at least 200m of useful connectivity with only one IEEE 802.11b access point. Further experiments will investigate possibilities to extend this range: firstly, by using external antennae for each single access point; secondly, by increasing the distance of access points further; and, thirdly, by changing access point parameters such as channel used and beacon transmission rate. Also, further reference measurements with different hardware components – access points and WLAN cards – and even with different firmware and driver versions have revealed significant performance differences (up to 50% UDP throughput!) for individual settings. These observations suggest careful selection and testing of products to obtain optimal (and comparable) results. Finally, implications from using 802.11a/g on the connectivity window (and particularly the total transfer volume) require further study.

VI. Drive-thru Internet and Applications

We have seen that WLANs may well enable Drive-thru Internet connectivity (in the sense of a nearlynet). We have also seen the connectivity periods and the – highly variable – transmission characteristics (data rate, packet loss rate, delay) for various measurement scenarios investigated. These are quite dissimilar from what we usually find on regularly loaded Internet access links, particularly regarding packet loss as well as the variations in transmission delay and throughput. While, obviously, further investigations are required, we will nevertheless provide a first interpretation of the observed characteristics in the light of typical Internet applications. Those may be affected in two ways: 1) by the limited connectivity period and 2) the observed variable transmission characteristics.

A. Intermittent Connectivity

Looking at current uses of the Internet, roughly two ways of user interactions can be distinguished: continuous communications such as IP telephony, database/file access, and the like on one hand and more transaction-based (request-response-style) information access, including email, data synchronization, and file sharing tools on the other. The former require largely persistent connectivity for the lifetime of the application instance (e.g. an IP phone call) and hence their usage is limited to the connectivity period – which may seriously limit their usability, at least in their present form. The latter may only need to complete individual transactions (e.g. sending an e-mail message) during one connectivity period and may continue their operation in the next connectivity window and hence are much better suited for this kind of environment. Web access and messaging/chat are examples for applications somewhere in-between, much dependent on the actual user behavior.

One approach to better support such applications – besides extending the connectivity period as far as possible – is to make them aware of upcoming connectivity islands so that they can start their transactions timely and to ensure that their network activity is aligned with the network signal quality, i.e. that intense data exchange takes place during the optimal transmission window (see also next subsection). To allow for transactions to complete quickly, the involved backend infrastructure (link to the service provider, servers, etc.) should be designed to not introduce additional unnecessary delays due to system or network load.[15]

A next step is to actually enhance existing applications and application infrastructures to better deal with intermittent connectivity. For example, the transaction-style applications may as well be able to operate in some kind of batch mode – where a user formulates a request or a series of requests that are dispatched at one connectivity islands and the results are retrieved at the same and possibly at the succeeding one(s). A local entity (e.g. a proxy cache for web applications) on the mobile device decouple the application (e.g. the user's standard web browser) from the Drive-thru network characteristics and thus conceal the intermittent nature of network connectivity. While this entity, the *Drive-thru client*, communicates via standard protocols with local applications, it uses some enhanced protocol to talk to a peer, *the Drive-thru proxy*, in the fixed network which is operated by a Drive-thru service provider. It uses the Drive-thru proxy as an intermediary to relay requests to e.g. web servers in the Internet and retrieve results even while the mobile device is not connected. Bundled results are forwarded to the Drive-thru client at the next opportunity. In summary, user and device mobility as well as recovery from frequent loss of connectivity is entirely

[15]The provider of Drive-thru connectivity islands obviously has only limited influence on the performance of third party email or web servers.

handled at some kind of shim layer between the transport and the application layer. The operation of Drive-thru client and Drive-thru proxy is conceptually comparable to some class of performance enhancing proxies (PEPs) used e.g. with Internet access through (one-way) satellite-based or cellular networks.

B. Transmission Characteristics

Irrespective of the type of application used, the transmission characteristics observed for Drive-thru network access influence the performance of the transport protocols underlying the applications. When passing through a connectivity island, link and IP layer connectivity is established fairly early – at a time when packet loss rates and transmission delays are still high. As our measurements show, the wireless access network does not become really usable until 150-200m (i.e. several seconds) later (entry phase) and remains so for at least 200m (production phase) before connectivity degrades again for another 150-200m (exit phase) until being entirely lost.

The significant delay and low throughput observed in the entry period may harm the initialization of transport protocols (e.g. for RTT estimation, determining network congestion, and calculating timeouts). Protocols such as TCP initially tune their parameters to match this low performance environment of the entry phase. While we have observed that TCP adjusts fairly quickly to improved link layer conditions when communicating with a peer located in the connectivity island, this adaptation may not work as fast when connected to a peer in the Internet with a much larger delay.

To counter these negative implications from differences in rapidly changing network characteristics, *TCP connection splitting* may be employed: the TCP connection from the mobile device is terminated at a local intermediary in the connectivity island and a separate connection is established for communications with hosts in the Internet. This approach isolates the changing WLAN environment and allows for efficient local adaptation as observed in our measurements. The intermediary, another type of performance enhancing proxy, may include additional functions of a Drive-thru proxy and thus further enhance communications.

UDP-based protocols may suffer from packet loss and thus may require forward error correction (FEC) schemes or appropriate application-layer repair mechanisms to be deployed. While our packet logs provide sufficient information to determine the impact of FEC, we have deliberately left these investigations for further study. Transport protocols for real-time media (such as RTP) also require proper error repair strategies to deal with losses and receivers; however, in particular, they need to tolerate highly variable delays, which may limit the value of interactive communications.

Regardless of the protocol in use, data transmission activity during the entry and exit phases will have a negative impact on other users in the same connectivity islands: reduced link layer data rate and repeated link layer retransmissions are likely to cut into the limited budget available for all users of the connectivity island. As a consequence, the communication activity of Drive-thru users should as much as possible be limited to the production phase: TCP connections should not be initiated before the entry phase has passed and UDP exchanges should not start earlier either.[16]

What should be carried out during the entry phase is autconfiguration (and possibly authentication) so that the mobile device is readily set up when the production phase is entered. The high delay and packet loss also affect dynamic configuration of the mobile device's IP addresses e.g. via DHCP, the establishment of IPsec or other tunnels, and DNS lookups, all of which are frequently needed in the initial stages of setting up communications. For example, DHCP retransmission timeouts are not optimized for low-delay operation in the presence of packet loss; further investigations are required to determine the actual impact the communication characteristics of the entry phase on the overall Drive-thru performance.

VII. Conclusion

In this paper, we have introduced the idea of Drive-thru Internet: the use of WLAN technology to provide network access for users traveling by car, particularly on highways or the autobahn. Using three different measurement settings, we have obtained reference parameters for our equipment, carried out proof-of-concept tests, and eventually validated the technical feasibility of our idea. Our measurements have shown that the coverage obtained from a single access point is much larger than expected, providing more than ten seconds connectivity even at speeds of 180 km/h. Using several access points to extend the reach of a connectivity island turns out to be more difficult and requires a larger distance between the access points or different parameterization than would be used for stationary users. We have also seen that the connectivity is – expectedly – poor at the edges of a connectivity island (entry and exit phase), with a negative impact on packet loss and transmission delay, but that over a distance of more than 200 meters network performance is excellent. We have managed to transmit a maximum of 9 Mbytes of data in a single pass through a connectivity island with a single access point which confirms the principal suitability of WLAN for Drive-thru networking.

After investigating the network characteristics with UDP-based test tools, we have analyzed the behavior of TCP in Drive-thru scenarios. We have identified different connectivity phases and have measured the impact for both UDP and TCP communication. One conclusion from our measurements is that TCP, despite abruptly changing network characteristics, performs reasonably well when used for connections between topologically close systems as it was the case for our test scenario.

Currently, we are primarily interested in understanding the transmission characteristics further – which is also reflected in the future work to be carried out in the near term: we will further investigate the use of several access points to

[16]This requires indications from the WLAN card driver about wireless networks and the respective signal strength so that e.g. a Drive-thru client can determine the link layer status without probing the network and initiate and suspend communications accordingly.

extend connectivity islands as well as placement of access points to improve signal strength (and thus network quality). In addition, we will investigate the characteristics of Drive-thru networking with other IEEE 802.11 variants, specifically IEEE 802.11g and IEEE 802.11a. Moreover, the different properties of 802.11 ad-hoc networking (without access points) with respect to the access-point-based setup described in this paper should be studied.

At the transport layer, our results suggest further TCP measurements, e.g., considering TCP connections with higher roundtrip times and with additional congestion and packet loss on transit networks. In addition, the effect of TCP enhancements such as TCP Fast Start [11] should be investigated. Besides further variations of our transmission parameters, we will include multicast transmissions in addition to point-to-point interactions. Transport and application aspects are already being considered by specific measurement tools that simulate different types of user interactions (file access, uploads, download, browsing) and provide reporting similar to *rtpspy*. One goal is to develop a model for Drive-thru scenarios that allows for simulations instead of the rather costly and time-consuming experiments on the autobahn.

The Drive-thru scenario is based on mobility of hosts that sporadically attach to network on the road. Further study in this area will have to address the question of mobility management. For example, the applicability of network layer mobility solutions such as Mobile IP and the applicability of enhancements such as Hierarchical Mobile IP [12] should be investigated. These solutions should be compared to other approaches for mobility support, e.g., application layer mobility. In section VI, we have outlined elements of an architecture to further support existing and future applications that manage mobility and may, to a limited extent, conceal the intermittent nature of connectivity in certain environments. Future research will flesh out these concepts in more detail and address operational issues such as autoconfiguration and WLAN access authorization as well as the specific support required at the transport and application layers.

Finally, Drive-thru Internet access is just one application example that leverages *intermittent connectivity* – an example that deliberately puts the emphasis on a highly dynamic environment. As our measurements have shown, even short periods of connectivity can achieve substantial information exchange that are likely to suffice for many transaction-style applications. With users moving slower, the connectivity periods may be prolonged but the user density may increase as well; these and other tradeoffs may be observable for any kind of setting. Overall, however, we expect the ideas presented here to be applicable to a broader range of application scenarios. Ultimately, every commuter using her laptop computer at home and in the office experiences intermittent connectivity – except that this happens at a rate that users and (application) protocols are used to deal with.

REFERENCES

[1] Clay Shirky, "Permanet, Nearlynet, and Wireless Data," http://shirky.com/writings/permanet.html, March 2003.

[2] Carsten Bormann and Niels Pollem, "Wbone: WLAN Roaming Based on Deep Security," TERENA Networking Conference (TNC), May 2003.

[3] Paramvir Bahl, Wilf Russell, Yi-Min Wang, Anand Balachandran, Geoffrey M. Voelker, and Allen Miu, "PAWNs: Satisfying the Need for Ubiquitous Secure Connectivity and Location Services," IEEE Wireless Communications, Vol 9, No 1, 2002.

[4] Simon Byers and Dave Kormann, "802.11b Access Point Mapping," Communications of the ACM, Vol 46, No 5, 2003.

[5] Kun-De Lin and Jin-Fu Chang, "Communications and Entertainment Onboard a High-Speed Public Transport System," IEEE Wireless Communications, Vol 9, No 1, February 2002.

[6] Mattias Esbjörnsson, Oskar Juhlin, and Mattias Östergren, "The Hocman Prototype - Fast Motor Bikers and Ad-hoc Networking," Proceedings of MUM, 2002.

[7] Jatinder Pal Singh, Nicholas Bambos, Bhaskar Srinivasan, and Detlef Clawin, "Wireless LAN Performance Under Varied Stress Conditions in Vehicular Traffic Scenarios," IEEE Vehicular Technology Conference Fall 2002, 2002, vol. 2.

[8] "Homepage of FleetNet," http://www.fleetnet.de/, 2003.

[9] Henning Schulzrine, Stephen Casner, Ron Frederick, and Van Jacobsen, *RTP: A Transport Protocol for Real-Time Applications*, July 2003, RFC 3550.

[10] IEEE, *Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, Institute of Electrical and Electronics Engineers, Inc., 1997.

[11] V. Padmanabhan and R. Katz, "TCP Fast Start: a Technique for Speeding up Web Transfers," 1998.

[12] Claude Castelluccia, "HMIPv6: A Hierarchical Mobile IPv6 Proposal," February 2000.