appeared in IEEE Transactions on Industrial Electronics, vol. 43, no. 6, pp. 1265-1282, December 2002

Measurements of a Wireless Link in an Industrial Environment using an IEEE 802.11-Compliant Physical Layer

Andreas Willig, *Member, IEEE*, Martin Kubisch, Christian Hoene, and Adam Wolisz, *Senior Member, IEEE*

Accepted for Publication
in IEEE Transactions on
Industrial Electronics

Abstract— **The design and simulation of coding schemes, medium access control, and link layer protocols for future industrial wireless local area networks can be supported by some understanding of the statistical properties of the bit error patterns delivered by a wireless link (which is an ensemble of transmitter, channel, receiver, modems). We present results of bit error measurements taken with an IEEE 802.11-compliant radio modem in an industrial environment. In addition to reporting the most important results, we draw some conclusions for the design of MAC and link layer protocols. Furthermore, we show that the popular Gilbert/Elliot model and a modified version of it are a useful tool for simulating bit errors on a wireless link, despite their simplicity and failure to match certain measured statistics.**

Keywords— **Wireless error measurements, industrial environment, IEEE 802.11 WLAN, stochastic bit error models, MAC Design**

I. INTRODUCTION

THERE is currently an increasing interest in making wireless transmission technologies available not only in offices and homes, but also in industrial environments. Two especially attractive features are the reduced need for cabling and the potential for truly mobile stations. However, due to the special constraints in industrial applications, e.g., hard real-time requirements, it is probably not the best solution to simply use existing wireless technologies and protocols, which often are designed for different application areas. Instead, we see the need to design and develop specialized protocols, specifically for medium access control (MAC) and link layer, which take both the characteristics of the wireless medium and the hard real-time requirements into account.

To design MAC and link layer protocols it is vital to have some understanding of the error patterns delivered by the *wireless link* (the notion of a wireless link is used as an abstraction of the ensemble of modems, transmitter, receiver, channel). The same protocol can show different behavior and performance for different error characteristics. For example, the results presented in references [29] and [28] show that bursty (Markovian) bit errors are beneficial for the performance of TCP, as compared to the case of independent errors with the same mean bit error rate. For the PROFIBUS (a popular fieldbus system in Europe) independent errors result in better delay performance and stability of the logical token passing ring than bursty errors [27]. Advance knowledge of the error characteristics can help the protocol designer to select appropriate protocol mechanisms, e.g., to choose suitable forward error correction (FEC) coding schemes or to find good rules on when to perform retransmissions.

It is often convenient for the protocol designer to evaluate protocols with simulations before developing a prototype implementation and performing complex measurements. A key part of such simulations are link error models, which, in principle, determine for a transmitted packet which of the receiver stations see bit errors; sometimes the exact position of errors within a packet is of interest. Often, simulation-based performance evaluations are done with *stochastic* link error models.[1] Here, a simple stochastic process which often can be described in terms of a few parameters, is employed to generate bit error patterns. Some models are frequently used in performance studies of MAC or link layer protocols, e.g. the *Gilbert/Elliot model.* It is beneficial to parameterize these models from "real data", obtained from measurements, or to use the measurement results as a motivation for developing better models.

To serve both needs, i.e., understanding of error patterns as input for MAC protocol design, and finding parameters for stochastic error models to make performance analyses more realistic, we have done measurements of a wireless links error characteristics. The study was taken in an industrial environment, to be specifically relevant for design and simulation of MAC protocols for wireless fieldbus systems. We have chosen to focus on a single scenario.

We have used an IEEE 802.11-compliant radio modem with direct sequence spread spectrum (DSSS) modulation. This choice was for several reasons: it is standardized, operates in the license-free 2.4 GHz ISM frequency band, it is widely used. Furthermore, hardware components are commercially available. Specifically, it was possible to obtain a chipset without any upper layer (MAC) functionality (Harris/Intersil PRISM I chipset as MACless version [1] [14]). When this study started, this chipset was very popular and used in commercial WLAN products.

Our measurement setup is built such that there is no

All authors are with the Telecommunication Networks Group (TKN), Department of Electrical Engineering, Technical University Berlin, Germany. E-mail: {willig,kubisch,wolisz}@ee.tu-berlin.de. A. Wolisz is also with GMD FOKUS.

This work was partially sponsored by the Deutsche Forschungsgemeinschaft (DFG).

[1] An alternative approach would be to use traces, but these are only rarely available and their handling is often perceived as clumsy.

bias introduced by upper layer protocols or operating systems. There is no MAC protocol nor any higher layer protocol, just a small engine for generating well-known packets. Hence, we have fine grained control over timing and content of the generated packets. But more important, using a MAC entity would have introduced undesired interaction with MAC mechanisms, e.g. packet discarding in case of wrong checksums or illegal MAC header fields.

Clearly, the study has its limitations. It likely cannot be simply extrapolated to other scenarios than the chosen one, or to other radio modems (like, e.g., the updated PRISM II chip set). One fundamental reason for this difficulty are the unique properties of wireless links in the 2.4 GHz range: radio waves can penetrate walls and are reflected by several materials. As a result, multiple copies of a signal may travel on several paths with different distances from transmitter to receiver. Errors occur not only due to noise, but also due to multipath fading. In addition, distance-dependent path loss, and co-/adjacent channel interference also influence the channel. Hence, the wave propagation environment (number of propagation paths, their respective loss) and its time-varying nature (moving people, moving machines) play a dominant role in constituting channel characteristics. In general it is far from being obvious or straightforward, how wave propagation characteristics or presence of noise / interference translate into error behaviour. This can be easily confirmed from experience with cellular phones, where sometimes there is only a small distance between seemingly good positions and bad positions. However, we think that with the scenario chosen some common characteristics of industrial environments are captured: presence of strong motors, many (sometimes moving) metal surfaces, moving people, and machines switching on and off. We believe that, although the quantitative results like mean bit error rates are likely not valid for other environments, the qualitative results (time-varying behaviour; presence, burstiness behaviour and order of magnitude of packet losses; high variability of error burst lengths) will carry over to similar environments, and are important for designing MAC protocols. This belief is confirmed by the fact, that certain qualititative results (regarding packet losses and time-variability) were also obtained in a similar study in an industrial environment [10], using a radio modem of a different manufacturer (see Section IX).

We present results of our study and discuss some implications for the design of MAC and link layer protocols for future industrial wireless local area networks (WLAN). The focus is on the statistics of the packet loss and bit error patterns delivered by the wireless link (via its interface provided by the *baseband processor*, see Section II-A) to the MAC- and link layer protocol. We do not try to "explain" our results in terms of physical properties of the environment (noise sources, propagation characteristics) or of the radio modem, since both were not completely accessible.

With respect to error modeling, we show that the popular Gilbert/Elliot model fails to match certain statistics of our measurements and propose a slight modification of the model. Nonetheless, when applied to an example system the Gilbert/Elliot model and its modification give quite good results in predicting selected protocol performance parameters. Therefore, we advocate these models as useful, yet improvable tools for simulating bit errors on a wireless link.

Our study is different from other bit- and packet level studies. As compared to [10], we put much attention on the statistics of bit errors and packet losses and their meaning for MAC design. Most other studies (summarized in Section IX) were taken in non-industrial environments, furthermore they incorporate a protocol stack with UDP packets over WaveLAN (CSMA/CA MAC protocol), which hides the reasons for unsuccessful packet delivery from the users. With our setup, the absence of any MAC- or higher level protocol does not only allow to have a close look at the input of any MAC protocol, it also allows us to assess more deeply the phenomenon of packet losses. This is, to our knowledge, not done in the literature (except from [10]), nor are they reflected in most protocol performance studies.

This paper is structured as follows: in Section II, we describe our measurement setup, while in Section III our approach to measurement evaluation is sketched. In Section IV, we discuss our goals and how these are addressed by chosing the environment and the set of fixed and variable parameters. In Section V, we present some baseline results, while in the following Section VI the issue of packet losses is investigated in some detail. The usage of simple stochastic models is considered in Section VII. In Section VIII we discuss basic consequences of our findings for the design of MAC protocols aiming at reliability. An overview of other bit- or packet level measurement studies is provided in Section IX and in Section X we conclude the paper. A more detailed presentation of the measurement results reported here along with additional material can be found in a technical report [25].

II. MEASUREMENT SETUP

In this section we give a brief overview of our measurement setup. More details can be found in [26] and [25].

A. IEEE 802.11 / PRISM I Radio Modem

In 1997, the IEEE 802.11 standard was finalized [21], describing a WLAN operating in the license-free 2.4 GHz ISM band (Industrial, Scientific and Medical band) and offering different bit rates: 1, 2, 5.5, and 11 MBit/s. The standard defines the physical layer (PHY, covering, e.g., modulation, spreading and the format of packets, see below) and the MAC layer (method of arbitrating access to the wireless channel), however, this work does not consider the MAC protocol. We have used a MAC-less radio modem (based on Harris/Intersil PRISM I chipset [1]), which is compliant with IEEE 802.11 and uses the direct sequence spread spectrum (DSSS) PHY. It offers amongst others the following modulation types/bitrates: 1 MBit/s with (D)BPSK modulation (Differential Binary Phase Shift Keying), 2 MBit/s with (D)QPSK (Differential Quaternary Phase Shift Keying), 5.5 MBit/s with CCK (Complementary Code Key-

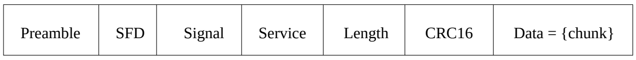

Fig. 1. Format of a packet

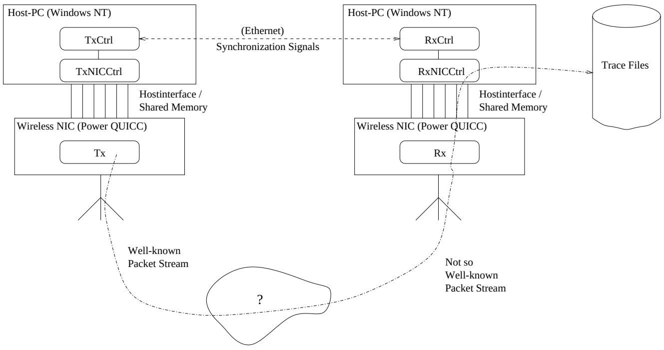

Fig. 2. Measurement Setup

ing), and 11 MBit/s with CCK modulation. Two antennas are attached to the modem to enable receiver diversity (i.e., the receiver selects the antenna with the maximum signal level). The transmitter power was fixed at 18 dBm, corresponding to 63 mWatt. The radio modem basically consists of high frequency circuitry and a baseband processor. The latter accepts and delivers a serial bit stream from upper layers, optionally scrambles the data (employing a shift register with feedback), and performs DSSS processing [14]. The characteristics of the serial bit stream is our focus of interest.

The baseband processor transmits and receives data in units of *packets*. A packet consists of a *header* and a *data part*, shown in Figure 1. The header fields control transmission and contain no MAC-related fields. A packet starts with a well-known preamble of fixed length, followed by a fixed value indicating the start of the header (start frame delimiter, SFD). The preamble and SFD allow the receiver to synchronize on the sender's clock (bit synchronization). The signal field indicates the modulation type used in the data portion of the packet, while the length field indicates the length of the data portion in microseconds (the service field has no significance). The CRC16 field contains a 16 bit cyclic redundancy check (CRC) checksum which is computed from the three previous values. If the checksum is wrong or the signal field carries an unknown value, the whole packet is discarded by the baseband processor. While the data part can use different modulation types, the header is always transmitted with BPSK modulation.

B. Measurement setup

We used two dedicated stations, a *transmitter station* and a *receiver station*, which do not change their roles during a measurement. The setup is sketched in Figure 2. The basic idea is that the transmitter station sends a well-known packet stream over the wireless link, which is captured and stored by the receiver station into a logfile. For generation and reception of the packets we use a microcontroller board carrying the radio modem and a separate processor (Motorola PowerQUICC [19] with PCI Interface and a 50 MHz PowerPC 603e processor). The coupling to the (Windows NT-based) host is achieved with a segment of 64 kByte shared memory, denoted as *host interface*. We call this board a *wireless NIC* (Network Interface Card). The wireless NIC contains a specific measurement application and neither MAC functionality nor any higher layer protocols. This way we have fine grained control over the packet generation and reception process and no bias is introduced by upper layer protocols.

Parameter	**Description**
$ScramblingEnabled$	determines whether scrambling is used
$DiversityEnabled$	determines whether receiver antenna diversity is used
$PreambleLength$	number of bits for PHY preamble
$ModulationCode$	distinguishes modulation used for data portion: 1 MBit/s BPSK, 2 MBit/s QPSK, 5.5 MBit/s CCK, 5.5 MBit/s BMBOK, 11 MBit/s CCK, 11 MBit/s QMBOK

TABLE I
ADJUSTABLE RADIO PARAMETERS

We briefly discuss the different software modules of our measurement setup, see Figure 2. The *Tx module* is located on the wireless NIC on the transmitter station. It accepts configuration commands from the TxNICCtrl module discussed below (allowing to set the variable parameters), and generates a well-known *packet stream*. The *Rx module* is also located on the wireless NIC. Its main task is to capture packets from the wireless link, to add metainformation (e.g., timestamps, packet size, reception status) and passing them to the host via the host interface (which puts them in a logfile). The resulting stream of received packets is denoted as *trace*. The *TxNICCtrl module* and *RxNICCtrl module* are wrappers, which offer a command line interface to the Tx module and Rx module. The *TxCtrl module* is a script which synchronizes itself with the RxCtrl software for controlling the measurements (using a TCP connection over the Ethernet). Finally, the *RxCtrl module* is actually controlling a whole measurement. It loops over all variable parameters; for each combination of parameters a packet stream is started (by triggering the TxCtrl module) and the trace is logged onto the harddisk. The evaluation of the traces is done off-line, employing several Perl scripts.

Our setup enables variation of several parameters, which are related both to the properties of the radio modem and packet stream generation. The important modem-related parameters are shown in Table I and the set of packet stream related parameters is shown in Table II.

Our setup was tested in laboratory measurements and in other measurement campaigns [13] in controlled environments and has shown to work fine. In several sometimes long running traces no bit errors, packet losses or other packet phenomena occured at all, or it was possible to relate the observed phenomena to environmental conditions.

Parameter	Description
NumPackets	Number of Packets
GapTime	Time gap between two packets
NumChunks	Number of chunks per packet, Packet length = *NumChunks* times 288 bits

TABLE II

Adjustable Packet Stream parameters

C. Format of the generated Packet Stream

The transmitter station generates a *packet stream.* On the other side, what the receiver captures is denoted as *trace.* When no errors occur, the trace is the same as the packet stream. The format of the packet stream was chosen such that: a) the number of 0's and 1's are equal; b) long runs of 0's or 1's are avoided; and c) it suffices to have a fraction of the packet (denoted as *chunk*) correctly received in order to determine which packet it originally was. Especially the last property enables bit-by-bit comparison of a received packet with the transmitted packet.

The generated packet stream consists of *NumPackets packets*, which are transmitted at equidistant start times, and all packets having the same parameters and packet size. The data part of a packet consists of an integral number of *chunks.* For generating a chunk, every bit of a 32 bit sequence number is mapped to eight bits (with $0 \mapsto 11000011$ and $1 \mapsto 00111100$), giving 256 bits. Additionally, 32 bits of header and trailer are generated, giving an overall chunk size of 288 bits. The sequence numbers are incremented from chunk to chunk. For example, with *NumChunks* = 3 chunks per packet, the first packet of a packet stream carries sequence numbers 0, 1, and 2, the second packet 3, 4, and 5 and so forth.

III. MEASUREMENT EVALUATION

Much of the evaluation of the measurements uses the notion of *indicator sequences* or the more special *binary indicator sequences.* First we give the according definitions, then we describe their use in measurement evaluation.

A. Indicator Sequences

In general, an indicator sequence is a finite sequence of natural numbers, a binary indicator sequence is restricted to the values zero and one. In the latter case, often we associate with a 1 an error event (e.g., an erroneous bit or a lost packet) and with a 0 the correct event. In this section, for convenience we will use the notion "bits" for the entries of a binary indicator sequence.

We subdivide binary indicator sequences into *error bursts* and *error free bursts* according to a *burst order* k_0. We define an error free burst of order k_0 to be a contiguous all-zero subsequence with a length of at least $k_0 + 1$. In contrast, an error burst of order k_0 is a subsequence with ones at its fringes, furthermore, within an error burst at most $k_0 - 1$ consecutive zeros are allowed.

By this definition a binary indicator sequence $i_1 i_2 \dots i_m$ of m bits length is segmented into p alternating error bursts and error free bursts. The length of the j-th error free burst is denoted as X_j, the length of the j-th error burst is denoted as Y_j and Z_j is the actual number of ones occuring in the j-th error burst. We write this as the *burst length sequence*[2]:

$$X_1, Y_1, Z_1 \quad X_2, Y_2, Z_2 \quad \dots \quad X_p, Y_p, Z_p$$

We denote the sequence $X_1 X_2 \dots X_p$ as the *error free burst length sequence*, $Y_1 Y_2 \dots Y_p$ as the *error burst length sequence* and $\frac{Z_1}{Y_1} \frac{Z_2}{Y_2} \dots \frac{Z_p}{Y_p}$ as the *error density sequence.*[3] It is important to note that with only recording the number Z_j of errors within error burst Y_j we loose information about the exact error positions (except fringes).

Using the notion of burst length sequences, some simple statistics can be computed, e.g. the mean error rate $\bar{e}$ or the mean error burst length $\bar{Y}$ by:

$$\bar{e} = \frac{\sum_{j=1}^{p} Z_j}{\sum_{j=1}^{p} (X_j + Y_j)}, \qquad \bar{Y} = \frac{1}{p} \sum_{j=1}^{p} Y_j.$$

Taking a binary indicator sequence $i_1 i_2 \dots i_m$ as a sequence of Bernoulli random variables, the conditional probability $\Pr[i_{n+k} = 1 | i_n = 1]$ for $(k \geq 1)$ is of some interest, since, assuming that $i_1, \dots, i_m$ have the same distribution and $\Pr[i_1 = 1] = \bar{e}$ is small, this probability approximates the correlation function $\mathrm{Corr}[i_n, i_{n+k}]$ of $i_1 \dots i_m$ (see [25, chap. 2]). It is approximated as follows (frequency-based approach):

$$\begin{aligned} \Pr[i_{n+k} = 1 | i_n = 1] \quad &\approx \quad \frac{\#\text{cases with } i_n = 1 \text{ and } i_{n+k} = 1}{\#\text{cases with } i_n = 1} \\ &= \quad \frac{\sum_{j=1}^{m-k} i_j \cdot i_{j+k}}{\sum_{j=1}^{p} Z_j}. \end{aligned}$$

B. Trace Evaluation

For every trace two important binary indicator sequences were computed. The *packet loss indicator sequence* (PLIS) of a single trace is constructed by marking lost packets with a 1 and received packets with a 0. During this step "suspicious" packets, such as ghost packets and bit shifted packets were identified and marked as lost packets (see Sections V-A and VI).

The *bit error indicator sequence* (BEIS) of a single trace is constructed by `XOR`ing every received (i.e., possibly erroneous) packet with its corresponding expected (error free) packet, and simply concatenating the results in the order of increasing packet numbers. Please note that in the BEIS

---

[2] We write $X_j = 0$ or $Y_j = 0$ to denote the absence of a burst at the fringes of a binary indicator sequence. Furthermore, we do not explicitly indicate the dependence on k_0 in the notation.

[3] As an example, take the binary indicator sequence 00100101000110001100000. With burst orders of $k_0 = 1$ and $k_0 = 2$ we get the burst length sequence

$$\begin{array}{lllllll} k_0 = 1: & 2,1,1 & 2,1,1 & 1,1,1 & 3,2,2 & 3,2,2 & 5,0,0; \\ k_0 = 2: & 2,1,1 & 2,3,2 & 3,2,2 & 3,2,2 & 5,0,0. & \end{array}$$

any information about packet boundaries, lost packets, or packet gap times is completely ignored.

The BEIS can be seen as the available input of a MAC protocol or a coding scheme.

While the PLIS in this paper is analyzed with $k_0 = 1$, for the BEIS several values of k_0 were used to get more insight into the burst structure.

IV. Measurement Parameters and Environment

We have conducted two measurement campaigns[4] in an industrial environment, namely at the Produktionstechnisches Zentrum (PTZ) in Berlin, Germany. The PTZ is a research facility for machinery engineering, supported by industry and academia. The first campaign was performed on June 26, 2000 and its main purpose was to evaluate our measurement setup and to find out which phenomena are important [26]. The second campaign took place from Aug. 28 to Aug. 30, 2000 [25]. In this paper, we focus on the second campaign.

A. Environment

The PTZ owns a large factory building, which contains several machines of different types and with people walking around all the time. The ground plan of the building has the shape of a circle. At the fringe of the circle is a path which can be used by small vehicles, while the inner circle contains the machinery. During both campaigns we have chosen the same positions within the building. The choice came from asking the PTZ people where they would place both stations. For a discussion of whether the chosen position was a "best case" or "worst case" position see Section X. In Figure 3, we show the relative position of our measurement equipment in the factory building, while in Figure 4 we show the close neighbourhood, especially the machines that are in close proximity. We have investigated a non line-of-sight (NLOS) scenario, with a closet in between the transmitter and receiver station, and the die sinking electrical discharge machine's (EDM) working area very close to the direct path. Both stations are ≈ 7-8 meters apart and not moved during the measurement campaigns. The receiver station was in close proximity (≈ 1 m) to a cabinet containing the power supply for a huge 5 axis milling machine, which, however, was not active during the second campaign. The die sinking EDM was active most of the time, except when changing the workpiece. A second EDM machine was situated behind the first one (see Figure 4). It was used by PTZ staff almost all the time. At the ceiling, in a height of ≈ 8 meters, was a portal crane, capable of moving around 20 tons. Its motors are placed at the fringe's end of the portal crane. The crane was used during the first two days of the second campaign.

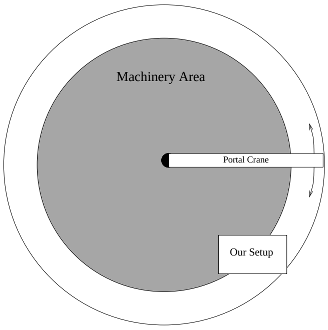

Fig. 3. Position of our Setup within the Building

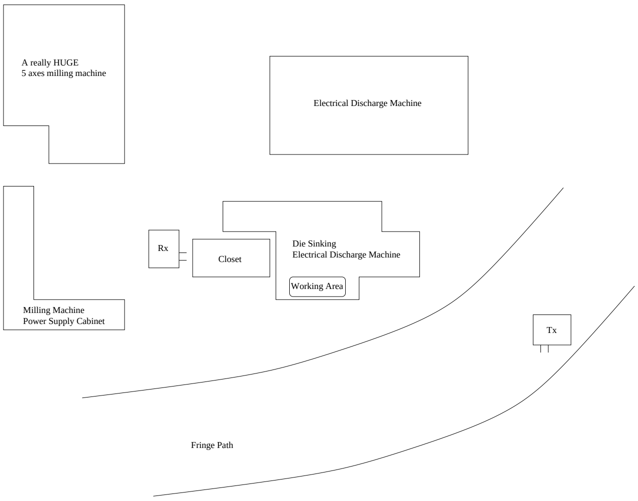

Fig. 4. Setup of PTZ measurement

Instead of investigating different scenarios with a restricted set of measurements, we have chosen to focus on the single scenario described above. The reason for choosing a NLOS scenario is that the measurement results should help in the design of MAC protocols and coding schemes for industrial WLANs, where we have hard requirements regarding timing behavior and reliability. Hence, it is interesting to see how MAC protocols and coding schemes react in the case of bad channel conditions, which can occur in reality.

B. Parameters

The second measurement campaign is designed to assess: a) the packet loss and packet impairment behavior on short and long timescales; b) the long-term bit error rate behavior; c) the dependency of the bit error behavior on packet sizes and modulation types; and d) the dependency of packet losses and impairments and bit error behavior on the scrambling mode. Furthermore, since we are interested in capturing the raw *link* behavior, we have assured that no explicit interferers in the same frequency band (e.g., IEEE 802.11 LANs) were present.

We have performed three different measurements within the second campaign: the **longterm1** measurement is a long-term measurement performed with a single modulation type and packet size, only varying the scrambling mode (addressing a), b) and d)). The **longterm2** measure-

---

[4] We give the following definitions: a *measurement campaign* consists of one or more *measurements*. A measurement is distinguished by the set of fixed and variable parameters from other measurements. Within a measurement, for each combination of parameters a packet stream is generated.

Parameter	**Value**
PreambleLength	128 bits
DiversityEnabled	True
Frequency	12
NumPackets	20000
NumChunks	14 (504 bytes)
GapTime	1000 μsec
ModulationCode	2 MBit/s QPSK

TABLE III

Fixed Parameters for **longterm1** and **longterm2** measurements

Parameter	**Value**
PreambleLength	128 bits
DiversityEnabled	True
Frequency	12
NumPackets	20000
GapTime	1000 μsec

TABLE IV

Fixed Parameters for **factorial** measurement

ment is the same as the **longterm1** measurement, however, another pair of PRISM I radio modems was used.[5] In the **factorial** measurement we have varied the scrambling mode, modulation type and packet sizes (thus addressing c)), and for each combination of parameters the short term bit error behavior was investigated. The main purpose of the **longterm2** measurement was to confirm that the observed phenomena are not due to the particular pair of radio modems used. Indeed, our results confirm this belief and allow us to restrict the discussion to the results obtained with the first pair of radio modems (the **longterm1** measurement and **factorial** measurement).

We have chosen for the **longterm1** and **longterm2** measurements to keep all parameters fixed, except the scrambling mode (on, off) and the pair of radio modems used (see Table III). In both measurements we have taken 90 traces for every scrambling mode. With 90 traces, 2 hours and 10 minutes are covered. Within a measurement the traces are numbered consecutively, thus the trace number corresponds to the time axis. Within the **longterm1** measurement the first 90 traces are taken without scrambling, the other 90 traces with scrambling. For the **factorial** measurement we have chosen a factorial design, varying the modulation type, packet size, and the scrambling mode as summarized in Table V, while keeping the other parameters fixed (Table IV). For every combination of parameters two traces are taken. The traces are numbered consecutively, i.e., in the order of their occurence. The traces 1 to 56 are taken without scrambling, the traces 57 to 112 with scrambling. Within each of the two groups we have varied the modulation scheme from low bitrates to high bitrates and for each modulation scheme we have varied the packet sizes from small packets to large packets.

Parameter	**Value**
ScramblingEnabled	True, False
ModulationCode	1 MBit/s BPSK, 2 MBit/s QPSK, 5.5 MBit/s CCK, 11 MBit/s CCK
NumChunks	3, 9, 14, 28, 56, 112, 167 (corresponding to 108, 324, 504, 1008, 2016, 4032, 6012 bytes)

TABLE V

Variable Parameters for **factorial** measurements

V. Baseline Results

In this section we describe the phenomena we have observed and present the most basic results on mean bit error rates and burst length statistics.

We exclude the 11 MBit/s CCK and 5.5 MBit/s CCK traces of the **factorial** measurement from further discussion, since these traces are extremely error prone. For example, in trace 52 (11 MBit/s CCK, 2016 bytes packet size, without scrambling) 19729 out of 20000 packets do not contain a single well-formed chunk.[6] Many of the 5.5 MBit/s CCK traces look also disastrous (specifically with scrambling) and thus are also excluded. A possible reason is that the noise level is too high.

A. Packet Related Phenomena

There are three major causes of transmission errors (see Sections II-A and II-B): Failure to acquire bit synchronization or to properly detect the start frame delimiter, an error in the header fields (e.g., wrong value in signal field or CRC error), and bit errors in the packet's data part.

Failing to acquire bit synchronization leads to *packet losses*. For the receiver station a lost packet is indistinguishable from the case that no packet was sent at all. Therefore, for detecting lost packets we inspect the timestamps of the packets in the logfile and compare them with the *InterpacketTime* (given as the sum of the *GapTime*, the fixed length header (see Figure 1) and the known length of the data part). Packet losses are discussed in Section VI.

An error in the header fields leads to other packet related phenomena (*ghost packets*, *missized packets* and *bit shifted packets*), which are discussed in [25, chap. 2, chap. 4]. The corresponding packets are treated as lost packets. This is reasonable, since their rates are low and they have the tendency to occur paired with lost packets.

Our setup is able to distinguish between getting no bit synchronization and the other packet phenomena, since the baseband processor generates an interrupt, when it has acquired bit synchronization and detected the SFD field. Therefore we can conclude that packet losses are due to not acquiring bit synchronization.

---

[5] The rationale for using a second pair of modems is to confirm that the effects visible are not due to a faulty pair of modems.

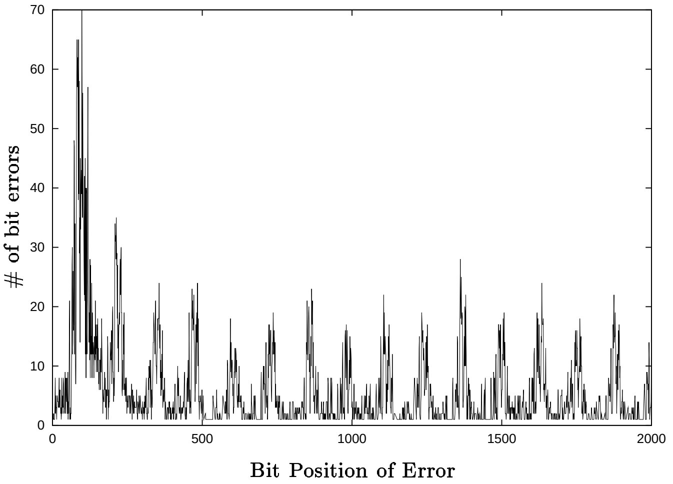

Fig. 5. Positions of Bit Errors, **factorial** trace 83 (QPSK modulation, with scrambling, 6012 bytes packet size)

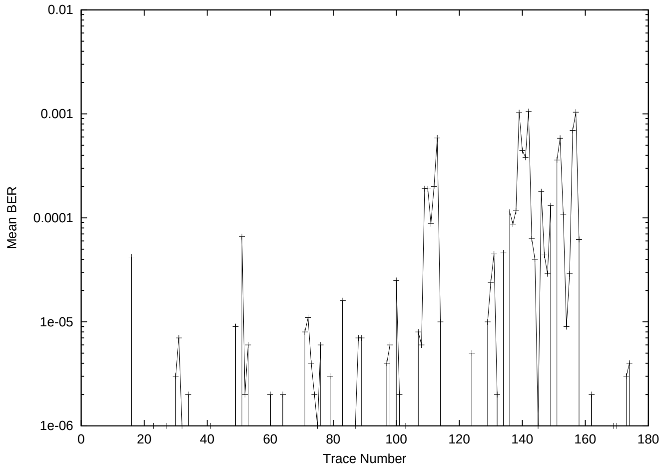

Fig. 6. Mean bit error rate vs. trace number for **longterm1** measurement (logarithmic scale)

Modulation	MBER w/o scrambl.	MBER w/ scrambl.
BPSK	2.5571e-05	0.0003
QPSK	7.4428e-05	0.0001
CCK (5.5 MBit/s)	0.0018	0.0399
CCK (11 MBit/s)	0.0544	0.0589

TABLE VI
MEAN BIT ERROR RATES FOR DIFFERENT MODULATION TYPES (**factorial** MEASUREMENT)

B. Positions of Bit Errors

Bit errors do not occur in all positions of a packet's data part with equal probability. This is exemplarily shown in Figure 5 for a QPSK trace of the **factorial** measurement. In the figure, for the first 2000 bit positions within a packet the number of bit errors occuring at this position during a trace is displayed. It is representative for the patterns occuring with QPSK.[7] The other modulation schemes also have their typical patterns, for BPSK they are similar to QPSK.

There is a peak at the beginning of a packet's data part. For QPSK and BPSK traces without scrambling it is frequently found between bit $\approx$ 200 and 250, for traces with scrambling often the peak is at positions $\approx$ 80 to 120.

The figures for BPSK and QPSK traces show some periodicity. From inspection, for BPSK traces the basic period is 64 bits, for QPSK it is 128 bits. This periodicity is visible with and without scrambling. In [25] we present some figures, which, for selected QPSK traces, show the conditional probability that bit $n+k$ is wrong given that bit n is wrong (calculated over the respective bit error indicator sequence (BEIS)). These figures indicate that indeed often bit errors have a distance of 128 bits (64 bits for BPSK). We have no validated explanation for this phenomenon, but after several personal discussions we think that it is due to artifacts of the wireless receiver's bit synchronization algorithm.

C. Mean Bit Error Rates

The mean bit error rates (MBER) per trace are varying over several orders of magnitude over time, even for the same modulation type and packet size. To illustrate this, we show in Figure 6 the MBERs for the **longterm1** measurement; please note that this figure spans more than four hours (see Section IV-B). It can be seen, that the MBERs are higher with scrambling enabled[8] (traces 91 to 180); our data backs this up. For the **factorial** measurement the MBER vs. trace# figure has the same characteristic of spanning several orders of magnitude.

From looking at the MBERs, only a few clear patterns emerge: MBERs seem to be higher with scrambling enabled, and furthermore, the MBERs increase with transmission speed, as is shown in Table VI. The BPSK modulation shows the best error rates, followed by the QPSK scheme. Other patterns, e.g. dependency on packet sizes, were not clearly visible; they are likely overshadowed by the inherently time-varying nature of the link.

D. Burst Length Statistics

As described in Section III, we have formed for every trace its bit error indicator sequence (BEIS) $i_1 i_2 \dots i_m$. With a given burst order k_0 we can investigate error bursts and error free bursts and their respective lengths $X_1 \dots X_p$ and $Y_1 \dots Y_p$.

In Figure 7 we show for selected BPSK traces mean error

---

[6] For many packets it was not possible to compute the error rate, since we could not determine the corresponding expected packet. For those packets where we could guess the expected packet, bit error rates of 25% to 30% are easily reached.

[7] Provided that we are looking at those traces where the number of errors is sufficiently high to have results of statistical significance.

[8] This is true for both the **factorial** and **longterm1** measurements, taken with the same modem set. For the **longterm2** measurement both scrambling modes show approximately the same mean bit error rate. A possible explanation goes as follows: the scrambler `XOR`s the received bit stream with the (already internally `XOR`ed) contents of a shift register of eight bits depth. The shift register changes its content with every bit in dependence of the incoming bit stream. Hence, an error in the bit stream propagates into the shift register and may influence the following eight bits.

burst lengths vs. k_0, while in Figure 8 the same is shown for selected QPSK traces. The respective curves are typical for BPSK and QPSK traces. In general, clearly the mean error burst length increases when increasing k_0 from k_0' to $k_0'' > k_0'$. The same is true for the mean error free burst lengths, since the error free bursts of length l with $l > k_0''$ survive as they are, while the error free bursts of length l with $k_0' < l < k_0''$ disappear and are not considered in mean burst length calculation. From Figure 8, for QPSK traces we can observe a trend to "step functions" with the steps having a distance of ≈ 128. After inspection of several traces it shows that this behavior is due to the periodicity of bit errors described in Section V-B. A similar behaviour, however, with a period length of 64, can be observed for BPSK traces (for other traces not shown in Figure 7 the "step function" character is more pronounced).

Of some interest is the view along the time axis: for selected values of k_0 we show in Figure 9 the mean error burst length vs. the trace number for the **longterm1** measurement. It can be seen that, even for all parameters fixed, the mean error burst lengths vary substantially over time (the mean error free burst length for fixed k_0 fluctuates over several orders of magnitude, not shown here). Furthermore, as already seen for the mean bit error rates in Figure 6, the mean error burst lengths are higher for scrambling enabled. Thus, the bit error characteristics vary not only on short timescales (from burst to burst, range of milliseconds), but also on larger timescales (trace order, range of minutes).

In Figures 10 and 11 we show for selected BPSK and QPSK traces of the **factorial** measurement and several values of k_0 the coefficients of variation (CoV) of the error burst length and error free burst length distribution of the respective BEIS. The variability of the error free burst length distributions is much higher than that of the error burst length distributions (typically between 1 and 3). This is due to very long periods of no errors within the respective BEIS. For the error free burst length distributions, there is a tendency of increased variability for increased packet sizes. This is likely due to the tendency of bit errors to cluster at the beginning of packets (see Section V-B): for large packets the packet beginnings have a larger distance in the BEIS, hence, likely longer error free bursts occur, which increases the variance. Furthermore, the CoV of the error free burst lengths is larger for BPSK as compared to QPSK. A likely reason is the typically lower MBER of BPSK traces, which leads to longer error free bursts, the latter increasing the variance. All observations are also true for the other BPSK and QPSK traces.

The bit errors for the BPSK and QPSK traces without scrambling have different characteristics. For BPSK the dominant case is that of single bit errors or cases where two single bit errors have a distance of 64 bits (see [25]). For QPSK we have different characteristics: many error bursts show two erroneous bits, and the bursts have lengths of 2, 4, 14 or 16 bits. Other burst lengths (and other numbers of errors in a burst) occur rarely.

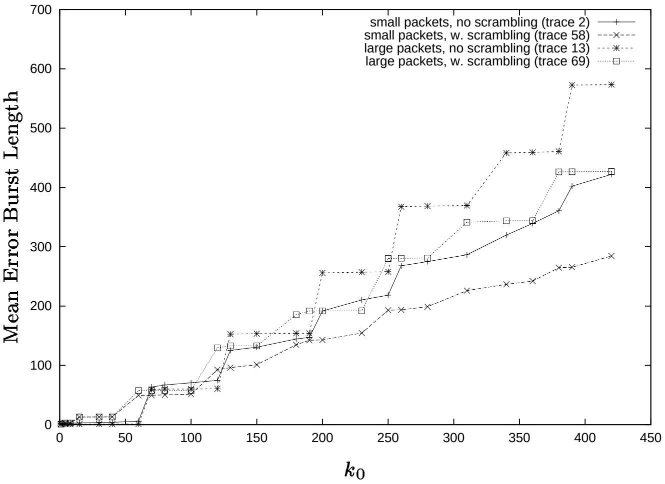

Fig. 7. Mean error burst length vs. k_0 for selected BPSK traces BEIS (**factorial** measurement)

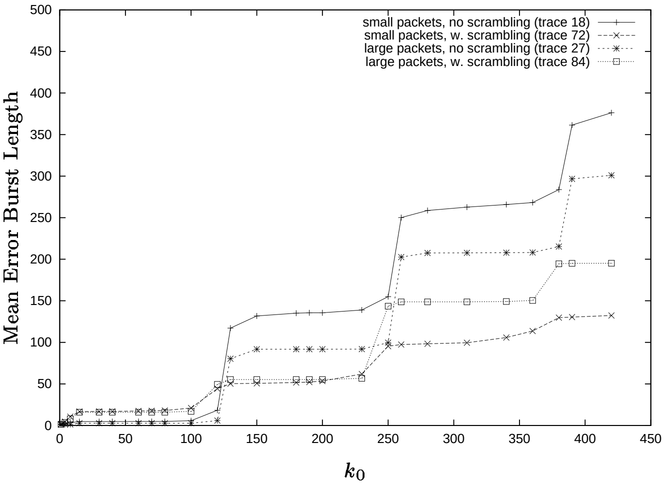

Fig. 8. Mean error burst length vs. k_0 for selected QPSK traces BEIS (**factorial** measurement)

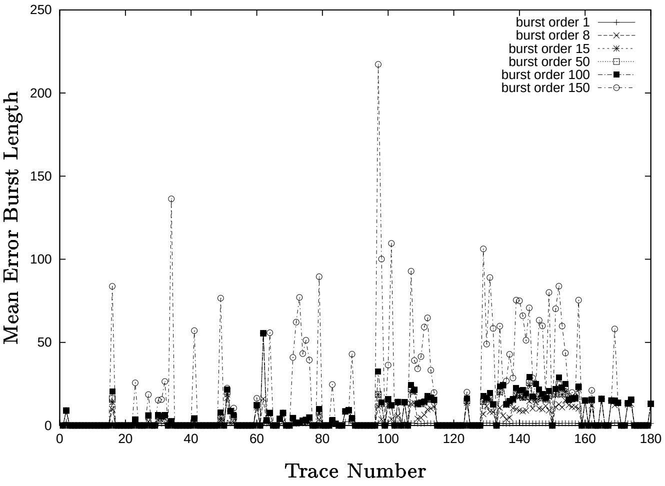

Fig. 9. Mean error burst length vs. trace number for selected k_0 (for BEIS of **longterm1** measurement)

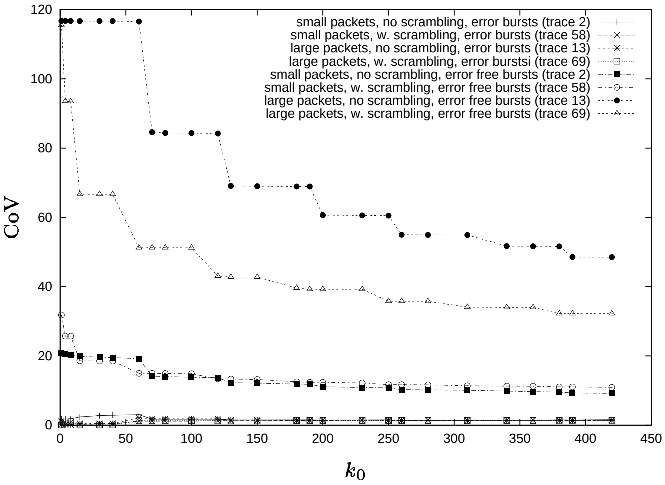

Fig. 10. Coefficients of Variation (CoV) of error burst lengths and error free burst lengths vs. k_0 for selected BPSK traces BEIS (**factorial** measurement)

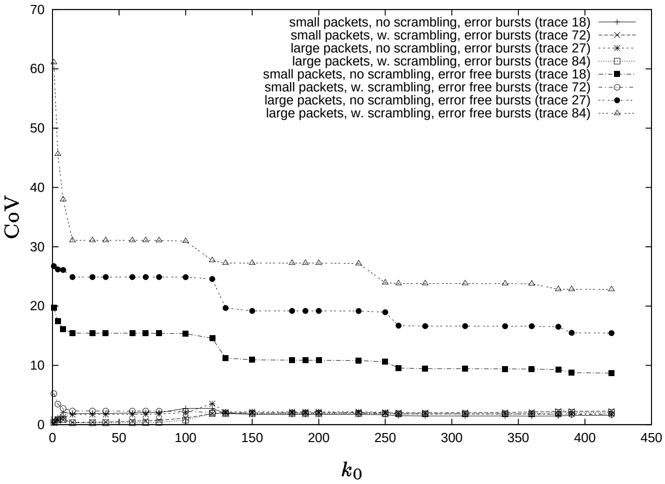

Fig. 11. Coefficients of Variation (CoV) of error burst lengths and error free burst lengths vs. k_0 for selected QPSK traces BEIS (**factorial** measurement)

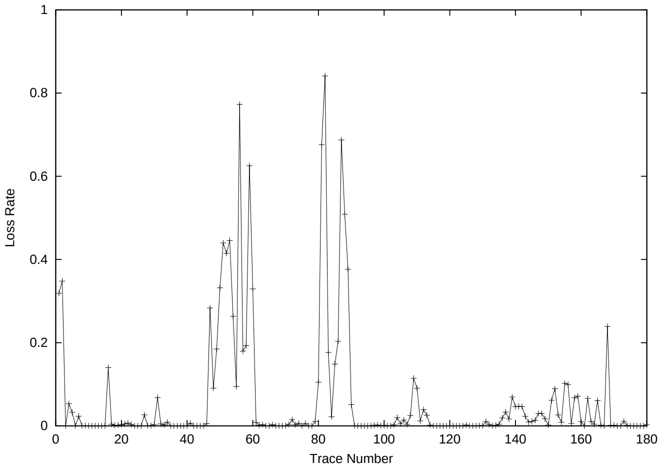

Fig. 12. Rates of lost packets for **longterm1** measurement

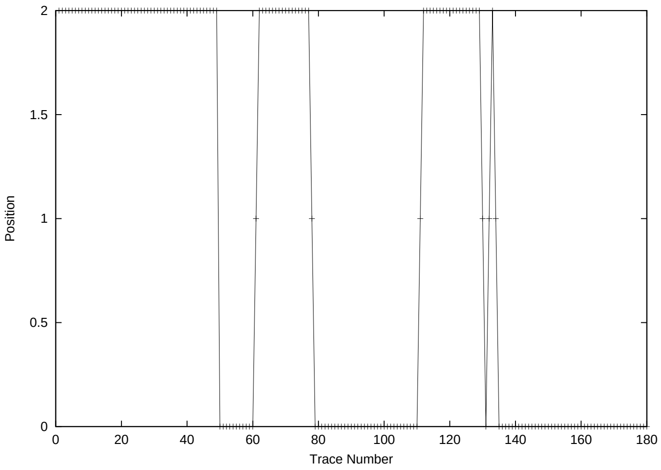

Fig. 13. Position of portal crane (0=close proximity, 1=short distance, 2=longer distance) for **longterm1**-measurement

VI. Packet Losses

In this section we discuss the packet loss behavior found in the **longterm1** measurement, as manifested in the corresponding packet loss indicator sequences (PLIS, see Section III-B). Instead of "error bursts" for the PLIS we use the term "packet loss bursts".

Packet loss rates are time-varying. To show this, we present in Figure 12 the rate of lost packets of a single trace vs. the trace number; please note that this figure spans over more than four hours (see Section IV-B). The packet loss rates are sometimes very high (more than 80%) and strongly varying. A possible explanation offers Figure 13, where the "portal crane function" for the **longterm1** measurement is shown. This function displays the distance of the portal crane to our setup (0 = directly above the setup, 1 = no more than five meters away, 2 = more than five meters away). It can be seen that, except for a peak at traces one and two, the packet loss rates have the highest values and the highest degree of fluctuation when the portal crane is close to the setup. During the **factorial** measurement the portal crane was not active and the packet loss rates are always below 10%.

We present some results on the "burstiness" of packet losses. To get summary information, we have formed the *compound packet loss indicator sequence* (COMP-PLIS) by concatenating the PLIS of all **longterm1** traces in order of increasing trace number. The COMP-PLIS was analyzed with burst order $k_0 = 1$. Hence, only consecutive packet losses belong to the same packet loss burst. Be $X_1 \ldots X_p$ and $Y_1 \ldots Y_p$ the corresponding packet loss free burst lengths and packet loss burst lengths. The main statistics of the packet loss burst lengths and packet loss free burst lengths are summarized in Table VII, and their respective distribution functions are shown in Figure 14. The overall packet loss rate is $\approx$ 6.4% and thus non-negligible. The packet loss bursts are typically short ($\approx$ 95% of all bursts last ten packets or less), but their lengths are highly variable, and very long bursts can be observed (long tailed distribution). The packet loss free burst length

Fraction of Received Packets	93.5144%
Fraction of Lost Packets	6.4855%
Received Packets: Mean BL	51.1254
Lost Packets: Mean BL	3.5457
Received Packets: CoV BL	20.0712
Lost Packets: CoV BL	17.1956
Received Packets: Max. BL	101158
Lost Packets: Max. BL	14936
Pr[Packet $n+1$ lost|Packet n lost]	0.7179
Pr[Packet $n+1$ received|Packet n received]	0.9804

TABLE VII

FIRST ORDER STATISTICS OF BURST LENGTHS (BL) OF THE **longterm1** MEASUREMENT (COMP-PLIS, $k_0 = 1$)

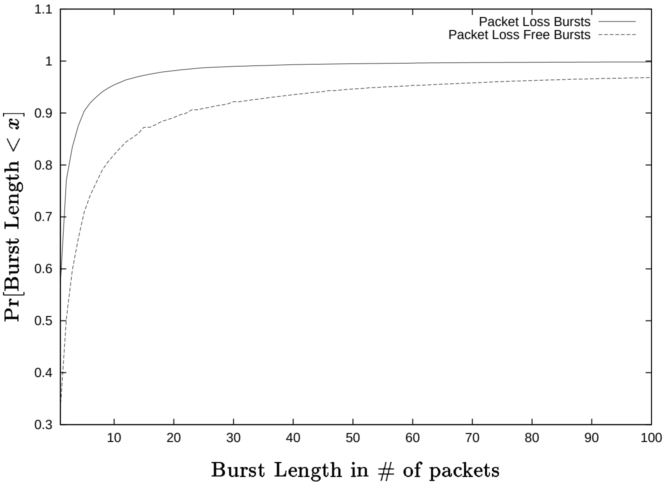

Fig. 14. Cumulative Distribution Functions of packet loss burst lengths and packet loss free burst lengths (for COMP-PLIS)

distribution is even more variable, has a higher mean value and a longer tail, which fortunately leads to long periods of no packet losses. Another view on the burstiness of packet losses is given by the conditional probability that packet $n+k$ is lost given that packet n is lost: it decays monotonically from ≈ 0.71 for $k = 1$ to ≈ 0.44 for $k = 2000$. Hence, by the approximation given in Section III-A, packet losses are strongly correlated over several hundreds of packets. With respect to the burst length sequences, it can be observed that the sequence $Y_1 Y_2 \dots Y_p$ is uncorrelated [25]. In contrast, the sequence $X_1 X_2 \dots X_p$ shows more than weak (> 0.2) correlation for short lags < 5 and weak correlation (< 0.2) for lags up to 100.

A surprising observation is documented in Table VIII, which shows the overall number of lost packets for the different measurements with and without scrambling. Packet losses occur significantly more often, if scrambling is disabled. Furthermore, not shown here, with scrambling the packet loss bursts are typically shorter than without scrambling.

	factorial	**longterm1**	**longterm2**
w scrambling	9885	34755	26392
w/o scrambling	20411	191456	45159

TABLE VIII

NUMBER OF LOST PACKETS

VII. FEASIBILITY OF SIMPLE BIT ERROR MODELS

Often the performance of proposed MAC and link layer protocols is evaluated with stochastic discrete event simulations, a key part of which are bit error models. Two popular models for wireless links are the *independent model* (where bit errors occur independently with a fixed rate p) and the *Gilbert/Elliot model* [12], [8]. The latter is quite simple and fails to match certain statistics of our measurements, but nonetheless gives surprisingly good results. We demonstrate this with a simple example system. Furthermore, the results can be improved with a small modification of the model. Our findings show that the Gilbert/Elliot model and its modification are a useful tool for simulating bit errors on a wireless link.

The Gilbert/Elliot model introduces two states named “good” and “bad” (in our setting these can be identified with the error-free bursts and error bursts, respectively). The intuition behind this is to capture the “bursty” nature of the channel, as observed in low-level measurements or predicted from propagation models. Conceptually, after every transmitted bit the new channel state is determined according to a discrete two-state time-homogeneous Markov chain. Within every state bit errors occur independently with rates e_g and e_b, respectively. A more detailed discussion of bit error models can be found in [25].

For the Gilbert/Elliot model, by the Markov property the state holding times are geometrically distributed and independent. When applying the model to our BEIS such that the mean burst lengths match, the resulting geometric distributions for both error burst lengths and error free burst lengths will have coefficients of variation close to 1. However, especially for the error free burst lengths coefficients of variations of 20 up to 100 are typical (see Figures 10 and 11), thus the “true” distributions are much more variable than the geometric distribution. In order to resolve this problem, we propose to use other distributions instead, and thus losing the Markov property. Therefore, we denote the resulting class of models as *semi-Markov models* [26]. Specifically, we use a quantized version of the lognormal distribution. It can be parameterized to match our traces burst lengths mean value and variance. Both models can capture short-term correlation for bit errors, but no long-term correlation, since in both models the burst lengths are independent. However, in our data for the error free burst lengths we often observed strong correlation for small lags and weak correlation for larger lags. For the error burst lengths we often have weak correlation for a longer time.

There are many other stochastic error models. Some models employing N-state markov chains are described in references [24], [9], [18], [15], [16], the class of *Hidden Markov Models* is treated in references [11] and [23], the class of *bipartite models* is introduced in [25].

	trace 24 ($k_0 : 100$)	trace 24 ($k_0 : 150$)
MBER	0.000370	0.000370
mean EBL	6.873	114.807
CoV EBL	2.457	2.341
max. EBL	229	6529
mean EFBL	6353.049	11514.112
CoV EFBL	77.441	57.775

TABLE IX
Summary statisitis of trace 24 (EBL=error burst length, EFBL=errorfree burst length, MBER=mean bit error rate)

Model	**Mean Time**	**Variance**
Trace	5915.63 s	0
independent	8028.61 s	0.47
Gilbert/Elliot	6100.30 s	0.65
Semi-Markov	5803.03 s	137.41
Null model	5540.00 s	0
Bipartite (20,20)	5917.76 s	608.74

TABLE X
Transmission Times for 1 GB data over channels with different error models (based on **factorial** trace 24)

A. Investigation of an Example System

One transmitter and one receiver station are connected via a wireless link. The transmitter wishes to transfer a file of 1 GB size to the receiver. The file is split into packets with the data part having the size of 1000 bits (the protocol overhead is neglected). There are no further stations present and we do not consider any MAC protocol or propagation delay. The transmitter sends a data packet and waits for an acknowledgement. If the ack does not arrive within two bit times the packet is repeated, otherwise the next packet is transmitted (Send and Wait Protocol). Data packets can be subject to errors, acknowledgements are always error free and of negligible size. The receiver only acks a packet if it contains no errors. The number of retransmissions per packet is unbounded. The transmission rate is 2 MBit/s QPSK.

For modeling the wireless link we have chosen to use trace 24 of the **factorial** measurement as the basis, its corresponding bit error indicator sequence (BEIS) was generated with burst order $k_0 = 150$ (the basic statistics are shown in Table IX). We have used the following error models: the independent model, the Gilbert/Elliot model, the semi-Markov model, a null model with no bit errors, a more complex 24-state bipartite model, and, finally, we have used the trace itself. We have not taken packet losses into account. The performance measure of interest is the time necessary to transmit the whole file, i.e. from sending the first packet until receiving the last acknowledgement. With the exception of the null model and the trace, every simulation was performed 40 times with different seeds of the pseudo random number generator.

The mean values reported in Table X show that indeed the Gilbert/Elliot model predicts the correct result with only 3.1% (Gilbert/Elliot) error, the semi-Markov model improves this to 1.9%. In contrast, the independent model generates an error of 35.7%. The results are quite good, despite the fact that all models fail to capture long-term correlation, as is demonstrated next.

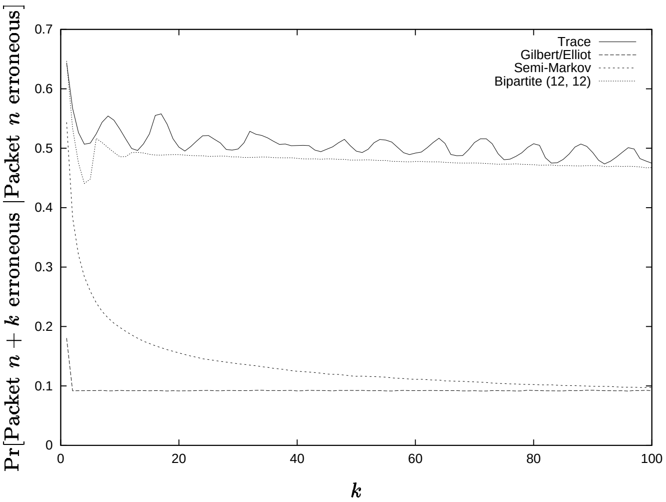

Fig. 15. Conditional Probability that packet $n+k$ is erroneous given that packet n is erroneous

From the receivers logfile we have computed the conditional probability that packet $n+k$ is erroneous, given that packet n is erroneous. The results are displayed in Figure 15. It can be seen that the Gilbert/Elliot model largely underestimates the correlation of the packet error behavior even on short timescales. The semi-Markov model gives better predictions on short timescales (< 5 packets), but then decays rapidly. The Gilbert/Elliot and semi-Markov models fail to predict the long-term correlation, instead they converge to their respective mean packet error rate of $\approx 10\%$ as taken from Figure 15. However, the "true" packet error rate is $\approx 6.3\%$. Hence, both models make significant failure in forecasting packet error rates. The possible influence of long-term correlation is an issue for further research.

VIII. Consequences for Protocol Design

Our results allow to draw some simple conclusions for the design of MAC and link layer protocols aiming at reliability. A general observation is that packet loss rates and mean bit error rates are time-variable, even for the same modulation type bit error rates vary over several orders of magnitude. This calls for inclusion of adaptivity into the protocol implementations. Possible control knobs are the transmit power, FEC code rate, number of retransmissions, modulation schemes, packet sizes, ARQ strategies, duplicating packets, and so forth. A MAC protocol can dynamically tune these parameters based on feedback and history information.

The occurence of (sometimes long lasting) packet losses is a challenge. Packet losses are due to failure of acquiring bit synchronization. This happens already in the header, thus no MAC protocol can protect itself against packet losses by influencing the contents of the data part of a

	Mean PER	**Max. PER**
BPSK w/o scrambling	0.9%	6%
BPSK w/ scrambling	6.4%	25.6%
QPSK w/o scrambling	7.7%	20.7%
QPSK w/ scrambling	3.2%	14.7%

TABLE XI

Mean Packet Error Rate (PER) and Max. PER for QPSK and BPSK modulation and different scrambling modes

packet. Instead it is necessary to incorporate other mechanisms, e.g., variation of transmit power level, using retransmission schemes, enable scrambling, or better shielding the radio equipment.[9] Furthermore, for invoking these mechanisms a feedback from the receiver is needed, i.e., the MAC protocol has to incorporate an immediate acknowledgement mechanism. Instead of using whole packets for immediate acks, it may suffice to rely on the presence or absence of short noise bursts / jam signals.

Consider a scenario, where a single base station (BS) serves a number of spatially distributed wireless terminals (WT), and the traffic is mainly from BS to WT and vice versa. Furthermore, we assume that channel access is time-multiplexed between stations, not code- or frequency multiplexed. In this case, and with multipath fading being a significant source of errors, the BS has for every WT a different propagation environment (number of paths and their respective loss). Hence, the BS has to every WT a separate channel, which can be viewed as independent from the others. If the errors on every channels have a bursty nature, this calls for introducing link state dependent scheduling approaches, as proposed in [3]. In this type of schemes the BS may decide to postpone retransmissions (triggered by a packet loss or packet error) and to serve another WT meanwhile. When assuming a strong FEC code and considering only packet losses, our results indicate that, if the retransmission is postponed for 5 to 10 packet times, with $\approx 95\%$ probability it will be successful (compare Figure 14).

The occurence of longer outage conditions due to packet losses should be recognized by a MAC protocol and signalled to upper layers to allow them to properly react (e.g., enabling emergency stop handling). In contrast to wireline or fibre optic communications, applications cannot assume the underlying network to be reliable, but should take changing network conditions explicitly into account. Therefore, some service primitives for signalling network conditions should be added to the interface of the MAC or link layer protocol.

The presence of long-term correlation in the packet error process (compare Figure 15 for an example) can be considered in different ways. If it is likely, that many of the packets following an erroneous packet will also be erroneous, it can be worthwhile to protect them by e.g. switching back to BPSK (increasing energy per bit), using FEC or to increase transmit power for a while. Another way is to use postponing schemes as sketched above for packet losses in a one BS / many WT scenario.

In [25, chap. 5] we investigate the feasibility of some simple block FEC schemes capable of correcting t bit errors in n ($n \in \{8, \ldots, 32\}$)[10] transmitted bits, which carry k ($k < n$) bits of user data. By the *Hamming bound*[11] [17, chap. 3] the best achievable code rate $\frac{k}{n}$ for $t = 1$ is $\approx 84\%$ ($(n, k, t) = (31, 26, 1)$), for $t = 2$ it is $\approx 71\%$ ($(n, k, t) = (31, 22, 2)$). Stated differently, if every packet is transmitted with FEC, we have an overhead of at least 16% for $t = 1$ and at least 29% for $t = 2$. In Table XI we show the mean and maximum packet error rate (all packets with at least one bit error) for BPSK and QPSK with and without scrambling.[12] If we consider the tendency of BPSK to have mostly single bit errors it suffices to look at codes with $t = 1$. For QPSK we often have longer bursts (see Section V-D) and we consider the case $t = 2$. Comparing the minimum redundancy needed for both cases with the mean and maximum PERs, and furthermore ignoring different packet sizes, we conclude that in every case it is wasteful to apply FEC to all packets. Instead, FEC should be enabled only for retransmissions, and, since packet errors show longer term correlation, FEC should stay enabled for a while.

A somewhat disappointing result is that the 5.5 MBit/s and 11 MBit/s CCK modulations are very sensitive and show disastrous results during our measurements. This finding is backed up by the results reported in [13], where under nearly ideal conditions (no interferers, line of sight, distance of 20 meters) the CCK modes also show unsatisfactorily results. Although it is common communications knowledge that higher modulation schemes are more susceptible to errors, the big jump in quality was surprising. Even for upcoming radio modem designs the different susceptibility of the modulation schemes is likely to remain. This suggests, that a good heuristic for protocol design might be to switch back to lower modulation schemes, when there are problems with the higher ones. For safety-critical data types (e.g., alarm messages) which tend to be short in industrial applications, it makes sense to always use the lower modulation schemes, since this increases probability of reception at only small costs (for short packets the PHY header takes most of the time).

IX. Related Work

In this section some other measurement studies are summarized, focusing on packet level or bit level measurements. For lower level (wave propagation) measurements of indoor scenarios see, e.g., reference [4] for measurements on channel impulse response, and reference [2] for an overview of

---

[9] At the time of writing it is not clear whether longer preambles would help.

[10] The restriction to 32 is somewhat arbitrary but can be justified by the observation that in industrial communications frequently we have very small packets, i.e. k is small.

[11] The Hamming bound states that up to t errors can be corrected in a codeword of n bits length and k user bits, only if the relation $2^{n-k} \geq \sum_{i=0}^{t} \binom{n}{i}$ holds. However, the fact that a triple (n, k, t) satisfies this relation, does not imply that a code with this properties really exists.

[12] One would expect increasing PERs for increasing packet sizes. This is only true for QPSK without scrambling.

propagation measurements and models. In the following, we restrict ourselves to indoor measurements.

Within the FUNBUS project some measurements with an IEEE 802.11-compliant DSSS PHY were carried out [10, chap. 9,10]. Namely, the Silver Data Stream radio modem [22] was used. Their measurement setup[13] has similarities to ours: MAC-less radio modem, dedicated transmitter and receiver stations, packet stream with equidistant start times and well-known packet contents (3, 64 and 252 bytes long frames). All measurements were performed without diversity, BPSK modulation, and scrambling enabled; the transmitter power was not given. Their setup was able to distinguish between lost packets, truncated packets (data part too short), erroneous packets (of correct length but with bit errors) and correct packets. In an outdoor LOS scenario their setup showed no transmission errors for distances up to 800 meters, hence it can be assumed to work properly. Amongst others, they have investigated an industrial scenario without interferers: the setup was placed in a hall of the University of Magdeburg with several (not further specified) machines in it. The authors assume a relatively large delay spread of 150 ns. There was no activity at the time of the measurements. The documented results indicate that in a LOS scenario for varying distance between 5% and 100% of all frames were error-free, however, there was no relationship to the distance. The missing packets are mainly due to packet losses (sometimes up to 60%) and truncations; the respective rates are varying. The transmission quality in NLOS scenarios was rated as "unusable". The presence and variability of packet losses was also confirmed in other scenarios. For example, when the two stations are placed at different positions within the same four-room apartment, no significant errors occured. However, when moving one station to another floor, very high packet loss rates (up to 100%) and packet truncation rates (up to 30%) could be observed, and the results are varying.

In a recent paper of Eckhardt and Steenkiste [7] adaptive error correction techniques are applied to WLAN traces, recorded in measurements using WaveLAN (902-928 MHz frequency band, 2 MBit/s QPSK modulation, receiver antenna diversity). They generate a specific UDP/IP packet stream, the underlying WaveLAN uses a CSMA/CA variant without retransmission on the MAC level. All packets are captured by the receiver, even if the WaveLAN checksum is wrong. The main findings are: a) bit errors are insensitive to the bit value; b) at short distances with no interferers the packet loss rate is zero and the packet error rate (PER, rate of packets with at least one bit error) is negligible, while with co-channel interferers the packet loss rates go up to 31%, a lot of truncated packets occur, and the PER is strongly varying. Almost all packets with corrupted bits have fewer than 5% of their bits corrupted. Bit errors do not have a trend to cluster in specific bit positions within a packet. Errors tend to occur in bursts, which are most often restricted to one or two bytes length (burst order $k_0 = 7$). The packet loss rate and bit error rate are insensitive to the packet size. The same authors have published another set of results on WaveLAN measurements before [6], using the same measurement setup. They have investigated signal quality parameters in an in-room line of sight scenario, a scenario with passive obstacles, and a third scenario with active interferers.

The work described in reference [20] concentrates on tracing and modeling of wireless channel errors on a packet level, incorporating a full UDP/IP protocol stack over WaveLAN (902-928 MHz frequency band, DSSS, QPSK, 2 MBit/s). All interference sources are suppressed. When only the load is varied (in terms of interarrival times for packets of fixed size 1400 bytes), the PER rate does not change. When varying the packet size, the PER doubles with every 300 byte increase of packet size, reaching $\approx 10^{-3}$ for 1400 bytes. When only varying the distance, the PER doubles every 17 feet, up to ≈ 0.08 at 130 feet. They defined a binary indicator sequence by assigning a one for an erroneous packet and a zero for an error free packet. For $k_0 = 1$ the mean error burst length was in most cases between two and three, while the mean error free length seems to decay almost linearly with increasing distance. The authors calculated suitable parameters for semi-Markov models for generating binary indicator sequences from their measurements.

One of the earliest WLAN packet-level studies is [5]. Again, a 902-928 MHz WaveLAN with 2 MBit/s QPSK, DSSS, and receiver antenna diversity was used. The authors have focused on varying the distance. For increasing distance the PER increases, however, there is a sharp cutoff, since it increases dramatically within a few meters, while before the increase rate was low. In their evaluation, if two erroneous bits occured in neighboured bytes, they belong to the same error burst. When evaluating their bit error indicator sequence (BEIS), they found that errors tend to be non-consecutive, typically only the minimum number of bits for constituting an error burst is erroneous (only one erroneous bit per byte). Furthermore, some error burst lengths are strongly preferred at all distances and packet sizes, e.g. 13 or 14 bits long. This is similar to our results with 14 or 16 bits long error bursts for QPSK. When looking at the burst length distribution functions, for longer runs they observed a (decaying) sawtooth pattern with maxima at multiples of 8. Hence, the authors also found some position dependency in the bit error behavior. The mean bit error rates are found to be "roughly constant" over all packet sizes and distances. An explanation for this could be that multipath fading instead of noise is the dominant source of errors. The effects of multipath fading do not correspond in a simple way to the distance.

X. CONCLUSIONS

In this paper we have presented the results of measurements done over a wireless link in an industrial environment.

The most obvious, yet far-reaching result is the variability of the wireless link over several timescales, even when

---

[13] developed at the ifak, Institute for Automation and Communication, Magdeburg/Germany

looking over hours. This concerns for example packet loss rates and mean bit error rates. We attribute this variability to frequently changing environmental conditons: moving people, portal crane activity, moving parts of machines, and so forth. Many industrial environments share this property of a frequently changing environment, hence, our study is representative in this respect. Stated differently: it cannot be said to represent "worst case" or "best case" conditions, but "typical" conditions (it must be understood that this claim refers to the input of a MAC protocol; regarding signal propagation it is hard to speak of "typical" conditions). This gives us some confidence, that, although one cannot directly transfer numerical results from this environment to others, the qualitative results in fact can be transferred: a) time-varying behaviour; b) occurence, burstiness properties, and orders of magnitude of packet losses; c) the great variability of error free burst length distributions for both packet losses and bit errors, leading to long periods of good conditions; d) the tendency of packet losses and bit errors (QPSK) to occur in bursts. Specifically the time-variability and the presence and orders of magnitude of packet losses was confirmed in a similar study. The same consideration applies to the restriction to a single scenario: when changing to a seemingly "better" position, likely this will not fix the variability. Furthermore, one often has not the freedom to move to "better positions".

We can learn several important lessons for the further development of industrial WLANs on the basis of the (Intersil PRISM I) IEEE 802.11 DSSS PHY. Beyond several statistical results, of some importance is the finding that the 5.5 MBit/s and 11 MBit/s modulation schemes showed serious performance problems. For the design of MAC and link layer protocols the phenomenon of (sometimes long lasting) packet losses is of utmost importance. More general, MAC and link layer protocols and coding schemes should incorporate some adaptivity, since the channel is time varying, both in terms of mean bit error rates and packet loss rates. And from a practical point of view, care should be taken in planning antenna locations.

The popular stochastic models used in the literature (independent model, Gilbert/Elliot model) fail to match the statistics of our data in several respects. Furthermore, it does not suffice to take only bit errors into account, but the phenomenon of packet losses should be modeled too. Nonetheless, the popular models give surprisingly good predictions for selected performance parameters of an example system, which can be improved by a slight modification of the model. We conclude that the Gilbert/Elliot model and its modification are a useful tool for simulating bit errors on a wireless link.

ACKNOWLEDGMENTS

We gratefully acknowledge the help of the group at the PTZ Berlin, namely Ulrich Doll, Hendrik Engel, Sascha Piltz, and Dirk Oberschmidt. Andreas Koepke from TKN was of great help in performing the simulations, Holger Karl and Morten Schläger gave insightful comments on earlier versions of the paper. The anonymous reviewers comments helped improving the paper. Finally we gratefully acknowledge the partial sponsorship given by the Deutsche Forschungsgemeinschaft (DFG).

REFERENCES

[1] Richard L. Abrahams. *2.4GHz 11Mbps MACless DSSS Radio HWB1151 Users Guide - AN9835.1.* Intersil, 1999.

[2] Jorgen Bach Andersen, Theodore S. Rappaport, and Susumu Yoshida. Propagation Measurements and Models for Wireless Communications Channels. *IEEE Communications Magazine*, 33(1):42–49, January 1995.

[3] Pravin Bhagwat, Partha Bhattacharya, Arvind Krishna, and Satish K. Tripathi. Using channel state dependent packet scheduling to improve TCP throughput over wireless LANs. *Wireless Networks*, 3(1):91–102, March 1997.

[4] Kenneth L. Blackard, Theodore S. Rappaport, and Charles W. Bostian. Measurements and models of radio frequency impulsive noise for indoor wireless communications. *IEEE Journal on Selected Areas in Communications*, 11(7):991–1001, September 1993.

[5] D. Duchamp and N.F.Reynolds. Measured performance of wireless lan. In *Proc. of 17th Conf. on Local Computer Networks, Minneapolis*, 1992.

[6] David Eckhard and Peter Steenkiste. Measurement and analysis of the error characteristics of an in-building wireless network. In *Proc. of ACM SIGCOMM'96 Conference,*, pages 243–254, Stanford University, California, August 1996.

[7] David A. Eckhardt and Peter Steenkiste. A trace-based evaluation of adaptive error correction for a wireless local area network. *MONET - Mobile Networks and Applications*, 4:273–287, 1999.

[8] E. O. Elliot. Estimates of error rates for codes on burst-noise channels. *Bell Systems Technical Journal*, 42:1977–1997, September 1963.

[9] B. D. Fritchman. A binary channel characterisation using partitioned markov chains. *IEEE Transactions on Information Theory*, 13(2):221–227, April 1967.

[10] Projektkonsortium Funbus. Das verbundprojekt drahtlose feldbusse im produktionsumfeld (funbus) – abschlußbericht. INTERBUS Club Deutschland e.V., Postf. 1108, 32817 Blomberg, Bestell-Nr: TNR 5121324, October 2000. http://www.softing.de/d/NEWS/Funbusbericht.pdf.

[11] J. Garcia-Frias and P. M. Crespo. Hidden Markov Models for burst error characterization in indoor radio channels. *IEEE Transactions on Vehicular Technology*, 46:1006–1020, November 1997.

[12] E. N. Gilbert. Capacity of a burst-noise channel. *Bell Systems Technical Journal*, 39:1253–1265, September 1960.

[13] James Gross, Michael Jaeger, and Andreas Willig. Measurements of a Wireless Link in different RF-isolated Environments. TKN Technical Report Series TKN-01-005, Telecommunication Networks Group, Technical University Berlin, June 2001. http://www-tkn.ee.tu-berlin.de/publications/tknrreports.html.

[14] Intersil. *HFA3860B Data Sheet, File Number 4594.1*, 1999.

[15] Young Yong Kim and San-qi Li. Modeling multipath fading channel dynamics for packet data performance analysis. In *Proc. IEEE INFOCOM 98*. IEEE, 1998.

[16] Young Yong Kim and San-qi Li. Capturing Important Statistics of a Fading/Shadowing Channel for Network Performance Analysis. *IEEE Journal on Selected Areas in Communications*, 17(5):888–901, May 1999.

[17] Shu Lin and Daniel J. Costello. *Error Control Coding – Fundamentals and Applications.* Prentice-Hall, Englewood Cliffs, New Jersey, 1983.

[18] R. H. McCullough. The binary regenerative channel. *Bell Systems Technical Journal*, 47:1713–1735, October 1968.

[19] Motorola Inc. *MPC 860 PowerQUICC Users's Manual*, 1998.

[20] Giao T. Nguyen, , Randy H. Katz, Brian Noble, , and Mahadev Satyanarayanan. A trace-based approach for modeling wireless channel behavior. In *Proceedings of the Winter Simulation Conference*, Coronado, CA, December 1996.

[21] The Editors of IEEE 802.11. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, November 1997.

[22] SRB Innovative Industrie Elektronik Gmbh. *Silver Data Stream Transfer-Modul – User Manual for SDSTM-F24-V1.0.1*, 1999.

[23] William Turin and Robert van Nobelen. Hidden Markov Modeling of Flat Fading Channels. *IEEE Journal on Selected Areas in Communications*, 16(9):1809–1817, December 1998.

[24] H.S. Wang and N. Moayeri. Finite State Markov Channel - A Useful Model for Radio Communication Channels. *IEEE Transactions on Vehicular Technology*, 44(1):163–171, February 1995.

[25] Andreas Willig, Martin Kubisch, and Adam Wolisz. Measurements and Stochastic Modeling of a Wireless Link in an Industrial Environment. TKN Technical Report Series TKN-01-001, Telecommunication Networks Group, Technical University Berlin, March 2001. http://www-tkn.ee.tu-berlin.de/publications/tknrreports.html.

[26] Andreas Willig, Martin Kubisch, and Adam Wolisz. Results of Bit Error Rate Measurements with an IEEE 802.11 compliant PHY. TKN Technical Report Series TKN-00-008, Telecommunication Networks Group, Technical University Berlin, November 2000. http://www-tkn.ee.tu-berlin.de/publications/tknrreports.html.

[27] Andreas Willig and Adam Wolisz. Ring Stability of the PROFIBUS Token Passing Protocol over Error Prone Links. *IEEE Transactions on Industrial Electronics*, 2001. accepted for publication.

[28] Michele Zorzi and Ramesh R. Rao. Performance of arq go-back-n protocol in markov channels with unreliable feedback. *Wireless Networks*, 2:183–193, 1997.

[29] Michele Zorzi and Ramesh R. Rao. Perspectives on the impact of error statistics on protocols for wireless networks. *IEEE Personal Communications*, 6(5), October 1999.

Andreas Willig (M '97) is currently a research assistant at the Telecommunication Networks Group at Technical University Berlin, working toward a Dr.-Ing. degree on wireless PROFIBUS systems. He holds a diploma degree in computer science from University of Bremen, Germany. His research interests include wireless and cellular networks, fieldbus systems, real-time systems and active networks.

Martin Kubisch is currently a research assistant at the Telecommunication Networks Group (TKN) at Technical University Berlin. He received his diploma degree in electrical engineering from Technical University Berlin in 2000 and his current research interests are protocol design w.r.t. energy aspects in a mobile environment.

Christian Hoene is a research assistant at TKN. Currently he is working on downloadable protocols and data link layer protocols for improved voice transmissions over wireless links. He received the Erwin-Stephan-Prize in 2000, awarded for fast and good studies. In 2001 he was with the COMET group, Columbia University, as a visiting scholar. His research interests include wireless networks, programmable networks, voice over IP and the development of data link protocols.

Adam Wolisz is currently a Professor of Electrical Engineering and Computer Science (secondary assignment) at the Technical University Berlin, where he is directing the Telecommunication Networks Group (TKN). Parallely, he is also member of the Senior Board of GMD Fokus, being especially in charge of the Competence Centers GLONE and TIP. His teaching activities encompass courses on Communication Networks and Protocols, High Speed Networks, Wireless Networks and Performance Analysis of Communication Networks. He is acting as a member of the Steering Commitee of the Computer Engineering Curriculum at the Technical University Berlin. He participates in the nationally (Deutsche Forschungsgemeinschaft) founded Graduate Course in Communication-Based Systems. His research interests are in architectures and protocols of communication networks as well as protocol engineering with impact on performance and Quality of Service aspects. Recently he is working mainly on mobile multimedia communication, with special regard to architectural aspects of network heterogeneity and integration of wireless networks in the internet.The research topics are usually investigated by a combination of simulation studies and real experiments. He has authored 2 books and authored or co-authored over 100 papers in technical journals and conference proceedings. He is Senior Member of IEEE, IEEE Communications Society (including the TCCC and TCPC) as well as GI/ITG Technical Committee on Communication and Distributed Systems.