

# The Adaptive-Intervals MAC Protocol for a Wireless PROFIBUS

Andreas WILLIG, MEMBER, IEEE, Andreas KÖPKE  
 Telecommunication Networks Group, Technical University Berlin  
 email: awillig@ieee.org, koepke@ft.ee.tu-berlin.de

**Abstract**—For the PROFIBUS, a standardized and well-known fieldbus system, it is attractive to use wireless media. A natural approach in creating such a system is to re-use as much existing technology as possible. In the area of wireless local area networks (WLANs) clearly the IEEE 802.11 standard is the leading technology. Hence, the question comes up, how this technology can be used in the creation of a wireless PROFIBUS. The PROFIBUS is used for its abilities in matching realtime requirements even under harsh environmental conditions. On the other hand, wireless links are error-prone and show time-variable behavior. Hence, an important question is, how good the so-called realtime performance, a set of measures taking reliability and timeliness into account, of the PROFIBUS protocol over wireless links is. This paper presents corresponding results and compares them with those of a new, polling-based class of protocols, called adaptive intervals. It shows up, that the adaptive intervals protocols achieve up to an order of magnitude better realtime performance than the PROFIBUS protocol. We conclude from this that it is a promising approach to replace the existing PROFIBUS MAC protocol by another protocol for creation of a wireless PROFIBUS.

## I. INTRODUCTION

The topic of a wireless fieldbus in general, and a wireless PROFIBUS in particular is up in the air, due to several benefits of wireless technologies, including mobility and reduced cabling need. A natural approach is to re-use as much existing technology as possible. In the area of wireless local area networks (WLANs) clearly the IEEE 802.11 standard is the leading technology [14], [16], [15]. Hence, the question comes up, how this technology can be used in the creation of a wireless PROFIBUS.

Fieldbuses are used for their abilities in matching realtime requirements even under harsh environmental conditions. However, wireless links are error-prone and show time-variable behavior, as is confirmed by error measurements of an 802.11 PHY in an industrial environment [22]. Hence, a question that comes up immediately is, how the PROFIBUS media access control (MAC) protocol behaves in case of a wireless link in terms of deadline misses. Clearly, the fewer deadline misses, the better. More precisely: the percentage of important (i.e., high priority) messages which can be transmitted reliably (i.e., acknowledged) within a prespecified time-bound should be as high as possible (e.g.,  $> 99.x\%$ ), even at the cost of other performance measures like throughput or mean delay. This criterion is in this paper more precisely defined as a set of performance measures, called the *realtime performance measures*. These measures capture a protocols

ability to deliver time-critical and safety-critical messages reliably over the error-prone wireless link. They provide the goal for optimization in the search of good MAC protocols for a certain class of wireless fieldbus systems, specifically the wireless PROFIBUS.

The PROFIBUS uses a token-passing protocol on top of a broadcast medium on the MAC level.<sup>1</sup> In this paper we show that the PROFIBUS MAC protocol delivers unsatisfying realtime performance. Hence, the question for alternative protocol approaches comes up. We present the *adaptive-intervals* MAC protocol, which uses a polling-based approach combined with a group testing feature [1] for improving the delay in low load situations. This protocol is targeted to replace the PROFIBUS MAC and link-layer protocol while offering the same link-layer interface to upper layers. The results in this paper show that this protocol delivers superior realtime performance under many circumstances. We conclude from this that it is a promising approach to replace the existing PROFIBUS MAC protocol by another protocol for creation of a wireless PROFIBUS.

This paper is structured as follows: in the following Section II we give some background information on the error behavior of wireless links and the PROFIBUS protocol. Furthermore, we introduce the realtime performance measures. In Section III we describe the adaptive intervals (AI) MAC protocol. In Section IV we report on the setup and the result of a simulation study done to assess the realtime performance of the AI protocol and the PROFIBUS protocol. Finally, after reviewing related work in Section V, we give our conclusions in Section VI.

## II. BACKGROUND

This section sketches the necessary background information. After a brief introduction to wireless links we discuss the PROFIBUS link layer services and the PROFIBUS MAC protocol. The link layer services are the necessary point of reference for the so-called *realtime performance measures*, a set of measures capturing reliability and timeliness constraints over error prone links.

### A. Wireless Links

Wireless links are typically error-prone and time-variable. It can happen that a link works rather good for ten minutes,

<sup>1</sup>This paper focuses on PROFIBUS-FMS, a PROFIBUS variant with multiple master stations.

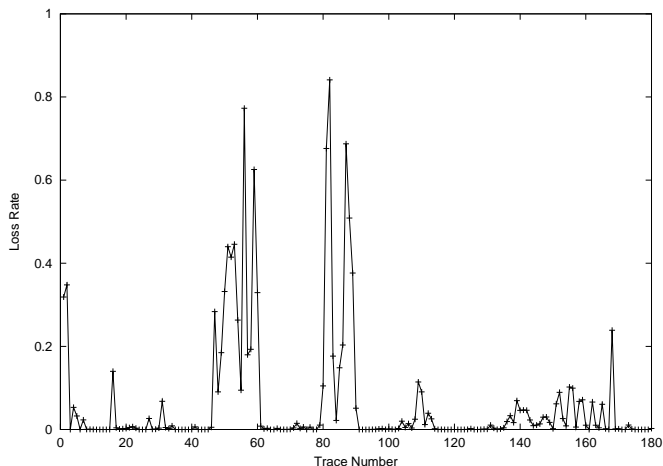


Fig. 1. Rates of lost packets for **longterm1** measurement (> 4 hours)

followed by being absolutely unusable for the next three minutes. Errors on wireless links occur due to different phenomena, namely multipath fading, path loss, co- and adjacent channel interference, man-made interference (microwave ovens, remote controls) and simple noise. Here, a wireless link refers to the ensemble of the actual channel and the wireless transceivers (including baseband processing, scrambling, DSSS processing, high frequency circuitry).

For the case of the popular IEEE 802.11 direct sequence spread spectrum (DSSS) physical layer (PHY) [14], [15] some measurements of the error characteristics of a wireless link in an industrial environment have been performed [22].

The link exhibits bit errors (change of bit values in a packets data part) and packet losses, other phenomena can be neglected. Packet losses are due to failure of acquiring bit synchronisation, and happen during the PHY preamble of a packet. For a receiver a lost packet is indistinguishable from a packet not sent at all. There is no way to influence packet loss behavior by any formatting rule for the MAC frame, since this only influences a packets data part, not the PHY preamble.

A very important finding is that both bit error rates and packet loss rates vary strongly over time. To illustrate this, we show some results obtained for a measurement spanning more than four hours. The measurement is subdivided into 180 so-called traces, in each trace 20000 packets of fixed size were sent. All traces have used the same set of parameters, which allows to compare them. The packet loss rate per trace is shown in Figure 1.

Bit errors and packet losses tend to occur in bursts. For packet losses, these bursts are with more than 90% probability shorter than ten packets, but occasionally very long bursts (more than 10000 packets in a row) have been observed (all with respect to packets sent at equidistant times). Stated differently: communication outages are typically short, but can occasionally last for tens of seconds or minutes.

## B. PROFIBUS

The PROFIBUS is a well known, standardized and widely used fieldbus [4], [17], [20]. The PROFIBUS standard covers

the application layer, the MAC- and data link layer and the PHY layer.

1) *Link Layer Services*: The PROFIBUS MAC- and link-layer protocol offers its services via the *link-layer interface*. There is one unacknowledged and three acknowledged services, in two of the acknowledged services the acknowledgement may carry some data. The acknowledged services are semi-reliable, i.e., a configurable number of retransmissions is performed, before the protocol gives up. The services are invoked by the upper layers using appropriate *request primitives* the link-layer interface offers. Once the outcome of a request is known (for the acknowledged services: an ack has received or the number of trials is exhausted) then the link-layer protocol generates a *confirm primitive* for the upper layers, indicating the success of the request. In this paper we focus on the SDA service (send data with acknowledgement), in which the acknowledgements do not carry data.

Each request is marked with one of two possible *priority values*. High priority requests are typically used for safety-critical and urgent messages (e.g., alarms), while low priority requests are used for everything else, including cyclic data transmission and file transfers.

## C. MAC- and Link-layer Protocol

The combined MAC- and link layer is called *fieldbus data link* (FDL) layer. On the MAC level two approaches are combined: a “request/answer” type of protocol for performing data exchange, and a token-passing protocol on top of a broadcast channel for arbitrating the right to initiate data or management transmissions. There are two types of stations: the *active stations* are capable of participating in the token passing protocol, the *passive stations* are not. The token is passed along a *logical ring*, which is formed by ascending station addresses. Station addresses are in the range of 0 to 126. Only ring members get the token and are allowed to initiate data transmissions. Every ring member carries out certain procedures for ring maintenance: detection and repair of lost tokens, including new stations into the ring, asking remote stations for their status, etc.

Upon every token arrival a station computes its token holding time according to a modified version of the timed-token protocol: a station subtracts its measured token rotation time from the configured *target token rotation time*  $T_{TTRT}$ . If the difference is greater than zero, then the station is allowed to initiate data transfers. However, even in the case of late tokens (difference smaller than zero) it is allowed to handle at least one high priority service request. If there is afterwards still token holding time available, the station handles the high priority requests first, then proceeds with the low priority request. A service request is processed according to a variant of the alternating bit protocol (ABP), with one separate protocol instance per target station. The number of retransmissions is bounded, the bound is a fixed parameter *max\_retry*.

New stations are included by stations already present in the ring: a ring member polls approximately every *gap\_factor* ·  $T_{TTRT}$  seconds the address space between its own address

and that of its logical ring successor. If a new station is found, it is included into the ring.

#### D. Realtime Performance Measures

The *realtime performance measures* are a set of measures targeted to capture the behavior regarding time and reliability. Due to the sometimes harsh and variable error conditions on wireless links, it is hopeless to give tight and deterministic guarantees on successful delivery of certain messages within a bounded time. Therefore, it is appropriate to express the requirements stochastically.

A set of  $N$  wireless stations is given, with each station running a MAC and link-layer protocol on top of a wireless medium, and offering the link-layer interface to a number of *request sources*. The service of interest here is the acknowledged datagram service without any data in the ack frames (SDA service).

The realtime performance measures are defined with respect to certain load models. For a single priority class, the term load is defined as follows: a load value of  $x$  percent has the meaning that in the case of no errors and without packets of other priorities present in the system, the load offered via the link-layer interface is such that the time spent for transmitting data frames of the given priority and including overhead is  $x$  percent of the theoretical link bandwidth. We assume always a fixed size of the requests, and for generating different loads the interarrival time of requests is varied.

In the *smooth* load model there is a low priority background load of  $x \in \{10, 50\}$  percent, which is splitted half into periodic arrivals and asynchronous arrivals (Poisson sources). For the high priority load Poisson arrivals with an overall load value of 10% are assumed. However, the sources generate high priority requests for a station only when there is no pending high priority request.

The main measure is delay-oriented. For a fixed station  $i$  the *confirmation delay*  $D_C(i)$  measure denotes the time between the arrival of a high priority request at the link-layer interface of station  $i$  and the time instant when the corresponding confirmation primitive is generated, i.e., the transmission outcome is known. As from the smooth scenario, high priority requests arrive always to a system with an empty high priority queue, hence, this measure doesn't take any additional queueing delay into account.

The  $D_C(i)$  measures are taken as random variables, and it is assumed that (due to the traffic scenarios chosen) all the values taken from the simulation belong to the same distribution. Denote for given station  $i$  by  $F_{D_C(i)}(x) = \Pr[D_C(i) \leq x]$  the distribution function of the confirmation delay values. The 99% percentile  $x_{99}(i)$  of this distribution is given by:

$$x_{99}(i) = \inf\{x \in \mathbb{R} : F_{D_C(i)}(x) \geq 0.99\}$$

We want to minimize the quantity  $\widetilde{D}_C$  (which we denote as *overall confirmation delay*):

$$\widetilde{D}_C := \max\{x_{99}(i) : i \in \{1, \dots, N\}\}$$

i.e. we want to minimize the maximum of the 99%  $D_C(i)$  percentiles over all stations.

As a side measure, it is also worthwhile to investigate the remaining bandwidth for low priority traffic  $B_L$  in the smooth scenario with 50% low priority load. Clearly, if two protocols show the same performance with respect to  $\widetilde{D}_C$ , the scheme which offers more bandwidth to low priority traffic (not to starve it out) is preferable.

Some additional remarks are in order:

- It is assumed that the maximum number of retransmissions is set to a high value  $\geq 20$ , in order to increase reliability. Hence, the *negative confirmation rate*, where the MAC/link-layer has to report about finally unacknowledged requests, should be close to zero. If this number is chosen to be a lower value (say, 3 to 5), then the negative confirmation rate would be of much interest.
- It is exactly the movement from a deterministic delay measure to 99% percentiles which accounts for the variability and error behavior of the wireless link, while simultaneously expressing hard delay requirements.

### III. ADAPTIVE-INTERVALS PROTOCOL

The basic framework for the *adaptive intervals* protocol is to the authors best knowledge not published so far. It belongs to the class of polling-based protocols, hence, it makes use of a central station / base station (BS). The basic idea is to divide the address range into groups and to run a group testing protocol [1] involving a random access slot and polling of the whole group in case of collisions. However, the groups are time-variable and change their size according to the perceived activity.

There are  $N$  stations having addresses  $1, \dots, N$ . At any time  $t$  the range  $1, \dots, N$  is partitioned into a number  $K(t)$  of subintervals  $[a_i, a_{i+1})$ . For an interval  $i$  the BS performs the following algorithm: if the size of the interval is larger than some threshold  $D$ , the BS sends a special frame (called RA slot announcement). This frame contains the interval bounds and announces to the stations lying in this interval, that the next transmission slot is a *random access* (RA) slot. Any of the stations in this interval having nonempty request queues sends a short announcement frame in the RA slot. Three things can happen: a) no station answers (this is denoted as *idle slot*); b) a single station answers and the BS receives the answer successfully (*transmission slot*); c) multiple stations answer or a single station answers but the BS does not receive the answer successfully (*collision slot*). In the first case the BS proceeds with the next interval  $i + 1$ , thus having used only a single RA slot to find out that more than  $D$  stations have nothing to send. In the second case the BS sends a poll frame to the successful station, allowing it to transmit some data. In the last case the BS sends a poll frame to every station in this interval, hence, allowing every station to transmit data. If the interval is smaller or equal than  $D$ , the BS simply polls all stations without using an RA slot.

So far the algorithm is similar to other group testing algorithms [1]. We propose to add some dynamics to this protocol and to let the intervals change over time. More specifically, we introduce certain heuristics of the form: if in two neighbored intervals no station answers in the RA slots a

certain number of times, these intervals are combined. Or: if collisions occur in the RA slot of an interval a certain number of times, this interval is splitted. These heuristics make use of history information which the BS gathers for every interval. More specifically, the BS maintains for every interval  $i$  three counters: the number of consecutively observed idle slots  $I_i$ , the number of consecutively observed collision slots  $C_i$  and the number of consecutively observed transmission slots  $T_i$ . A heuristic can be formulated with certain thresholds: if an interval  $i$ 's collision counter  $C_i$  is larger than a constant  $C$  and the interval is larger than  $D$ , the interval is splitted. Or: if two neighbored intervals show idle counters larger than  $I$ , they are combined. There are many degrees of freedom in selecting the threshold values and also initial values for the newly constructed intervals.

This approach aims to adapt to the load. The hope is that unloaded stations get polled less and less often and that the delays for loaded stations reduce. For high load situations the protocol converges to a round-robin polling.

We investigate three versions of this protocol. The basic version (called ai algorithm) works as described before. The ai-hp-lp algorithm runs two separate instances of the protocol for the high and low priorities. To enable this, the RA slot announcement frame indicates the priority value. If the indicated priority is high, only those stations may answer in the RA slot, that have a high priority frame available. In the ai-hp-rr algorithm the adaptive interval scheme is only executed for high priorities, and intertwined with it a round-robin scheme is executed.

#### IV. RESULTS

We have performed extensive simulations of the PROFIBUS protocol and the different adaptive interval protocols. For the PROFIBUS protocol we have enabled the enhancements discussed in [23], which are targeted at preventing certain scenarios for destruction of the logical PROFIBUS ring and for re-including lost stations faster.

For the PROFIBUS protocol, for any given number of stations  $N$  several simulations were carried out, with different values for certain protocol parameters (target token rotation time  $T_{TTRT}$  and gap factor). The values for these protocol parameters are chosen such that stations lost from the ring (due to lost or erroneous token frames) are re-included as fast as possible; they are not typical "real-world" values, since in existing wired plants including a station into the ring is assumed to be a rare event, not worth spending much bandwidth. The simulation parameters are summarized in Table I.

In the simulations the wireless link is assumed to show both bit errors and packet losses. Both kinds of errors were modeled according to a popular stochastic error model for wireless links, namely the Gilbert-/Elliot model [5], [2]. This model is based on a two-state Markov chain for each of bit errors and packet losses. In the *good* state the channel shows no bit errors or packet losses, in the *bad* state bit errors / packet losses occur independently with a certain rate. Due to the Markovian nature the state holding times are geometrically distributed.

| Parameter              | Values                        |
|------------------------|-------------------------------|
| Number of stations $N$ | 2, 4, 8, 12, 16, 20           |
| $T_{TTRT}$             | 0.005, 0.01, 0.015, 0.02, 0.4 |
| gap factor             | 1, 2, 4, 6, 8                 |
| $max\_retry$ parameter | 20                            |
| Bitrate                | 2 MBit/s QPSK                 |

TABLE I  
SIMULATION PARAMETERS FOR COMPARISON BETWEEN PROFIBUS AND THE AI PROTOCOLS

The model is parameterized from the traces reported in [22]. The overall mean bit error rate is 0.000370, the overall packet loss rate is 0.1099.

The propagation delay is zero, both simulators use 2 MBit/s QPSK modulation. The simulation time was always chosen to be 1000 simulated seconds.

In Figure 2 we show the  $\widetilde{D}_C$  performance for the different versions of the adaptive intervals protocol and the PROFIBUS protocol for the Gilbert-Elliot error model and a low priority load of 10%. The PROFIBUS value is determined as follows: for a fixed station number  $N$  take the minimum  $\widetilde{D}_C$  value for all possible gap factors and  $T_{TTRT}$  values, i.e., the best achievable value for the given parameter ranges. In Figure 2 the same is shown for 50% low priority load.

It can be seen that all adaptive interval algorithms clearly outperform the PROFIBUS protocol for the Gilbert-Elliot error model with respect to  $\widetilde{D}_C$  performance, which is also true for other bursty error models, but not for an error model with independent errors for both bit errors and packet losses (not shown here). The gain is up to an order of magnitude. An explanation is the ring instability of PROFIBUS [23]: due to repeated losses of token frames it can happen that stations get lost from the logical ring. Any pending high priority request of lost stations have to wait until the station is re-included into the ring. However, this can take significant time. Ring stability is a much more serious problem for bursty errors, where the channel is in a bad state for some time, making failed token passing more likely. For the case of independent errors the consecutive loss of a token frame and its repetitions is not very likely. Hence, ring stability is much better for independent errors.

For the 10% low priority load case all adaptive interval algorithms show approximately the same  $\widetilde{D}_C$  performance. For the 50% case the extra measures for high priority requests in the ai-hp-lp and ai-hp-rr protocols pay out by giving up to 35% gain compared to the ai algorithm.

#### V. RELATED WORK

There exists much literature on polling systems in general, mostly concerned with their queueing analysis [19], [18].

The topic of packet polling systems with transmission errors is treated rarely in the literature. In [19] some results for infinite buffer polling systems with Bernoulli feedback are presented. Under Bernoulli feedback for every queue the central station serves the head of the queue. Then a Bernoulli experiment is performed and the customer leaves the system

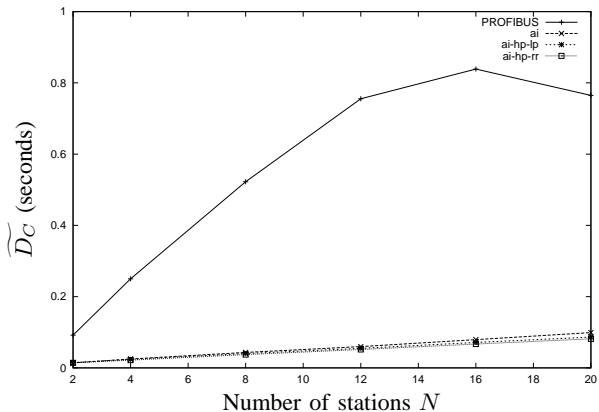


Fig. 2. Overall confirmation delays  $\widetilde{D}_C$  for the adaptive interval algorithms and the PROFIBUS protocol vs. number of wireless terminals  $N$  for 10% low priority load and the Gilbert-/Elliot error model

with some fixed probability or stays at the head of the queue. In [24] a polling system with only downlink traffic (requested by the wireless stations) is investigated under the Gilbert-Elliot error model. The focus is on the efficiency of the protocol under high loads. Deadline misses are not considered. In [6] a table based polling protocol is investigated under the Gilbert-Elliot error model. The protocol performs retransmissions, hence uses feedback. The polling table accommodates synchronous and asynchronous traffic. The mean response time and bandwidth utilization vs. load are investigated using a simulation approach.

The topic of wireless fieldbus systems is treated rarely in the literature. Some selected references for other fieldbus systems include [12], where a group at EPFL Lausanne has reported work on an application layer gateway for integrating wireless stations into FIP, employing a TDMA scheme on the wireless side. In [13] it is investigated, how the MAP/MMS protocol can be enhanced with mobility. In the proposed system the IEEE 802.11 MAC protocol with the (stochastic) DCF is used, time critical transmissions are not considered. In [11] the same question was investigated with DECT as underlying technology. Again, time critical transmissions are not considered. For the CAN fieldbus different mappings of the CAN MAC protocol to wireless media are discussed in [9], [8], however, channel errors and retransmissions were not taken into account.

The Funbus project [3], [7] was an industry-driven project with the goal of finding a suitable technology for wireless and transparent coupling of field devices. Within the project three different fieldbus technologies (PROFIBUS-DP, INTERBUS-S and CAN) are investigated with focus on IEEE 802.11 DSSS technology. The successor project R-FIELDDBUS ([www.rfieldbus.de](http://www.rfieldbus.de)) evaluates different wireless technologies for the PROFIBUS, P-NET and WorldFIP fieldbus systems with focus on multimedia support. In [10] a scheme for integration of wireless nodes in a PROFIBUS-DP network (single master, many slaves, no token passing) is described. The approach is to use an application layer gateway, which is integrated with a “virtual master”

The first author has recently proposed other types of polling-

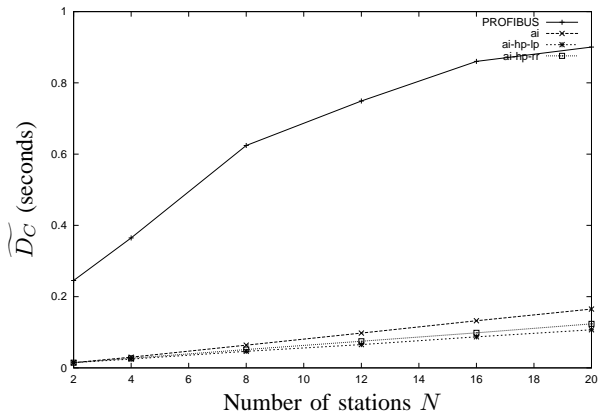


Fig. 3. Overall confirmation delays  $\widetilde{D}_C$  for the adaptive interval algorithms and the PROFIBUS protocol vs. number of wireless terminals  $N$  for 50% low priority load and the Gilbert-/Elliot error model

based protocols for a wireless PROFIBUS [21]. These show the same qualitative behavior as the adaptive interval protocols: for bursty errors they have significantly better realtime performance than the PROFIBUS, which changes for independent errors.

## VI. CONCLUSIONS

The results of this paper show that for the creation of a wireless PROFIBUS, subjected to error-prone and time-variable wireless links, the approach to replace the existing PROFIBUS MAC protocol by another protocol is promising. Actually, the PROFIBUS protocol suffers from stability problems with the logical token passing ring. With the protocol described in this paper, a gain in  $\widetilde{D}_C$  performance of up to an order of magnitude can be achieved.

However, with respect to a wireless PROFIBUS system, it does not suffice to find a reasonable MAC protocol. First, we believe that the link layer protocol is also an area of improvement, e.g. by introducing preemptability of low priority requests. The second, very important point comes from the wish to run wireless and wired PROFIBUS stations in a single PROFIBUS LAN. In this case two different MAC/link layer protocols are used, which need to be carefully integrated.

We believe that this paper can serve as a good starting point for further research in wireless MAC protocols with good realtime performance over wireless links. The existing fieldbus protocols, specifically the PROFIBUS, are often not designed with a very lossy link in mind. We see the need to look for protocols, which take the error characteristics of wireless links explicitly into account. With respect to the adaptive interval protocols, there are several interesting issues for further research. One example is to investigate the protocol for the case of uneven load patterns, where few stations generate much load and many stations only little load.

## REFERENCES

- [1] J. I. Capetanakis. Tree Algorithm for Packet Broadcast Channels. *IEEE Transactions on Information Theory*, 25(5):505–515, September 1979.
- [2] E. O. Elliot. Estimates of error rates for codes on burst-noise channels. *Bell Systems Technical Journal*, 42:1977–1997, September 1963.

- [3] 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>.
- [4] German Institute of Standardization (DIN). *PROFIBUS Standard Part 1 and 2*, 1991.
- [5] E. N. Gilbert. Capacity of a burst-noise channel. *Bell Systems Technical Journal*, 39:1253–1265, September 1960.
- [6] A. Hoffmann, R. J. Haines, and A. H. Aghvami. Performance analysis of a token based mac protocol with asymmetric polling strategy ('topo') for indoor radio local area networks under channel outage conditions. In *Proc. International Conference on Communications (ICC)*, pages 1306–1311, New Orleans, Louisiana, 1994. IEEE.
- [7] L. Rauchhaupt Jörg Hähnliche. Opportunities and problems of wireless fi eldbus extensions. In *Proc. FeT'99: Feldbustechnik – Fieldbus Technology*, Magdeburg, 1999. Springer Verlag, Wien / New York.
- [8] A. Kutlu, H. Ekiz, M. D. Baba, and E. T. Powner. Implementation of "comb" based wireless access method for control area network. In *Proc. 11th Intl. Symp. on Computer and Information Science*, pages 565–573, Antalaya, Turkey, November 1996.
- [9] A. Kutlu, H. Ekiz, and E. T. Powner. Performance analysis of MAC protocols for wireless control area network. In *Proc. Intl. Symp. on Parallel Architectures, Algorithms and Networks*, pages 494–499, Beijing, China, June 1996.
- [10] Kyung Chang Lee and Suk Lee. Integrated network of profi bus-dp and ieee 802.11 wireless lan with hard real-time requirement. In *Proc. IEEE 2001 International Symposium on Industrial Electronics*, Korea, 2001. IEEE.
- [11] Philip Morel. Mobility in map networks using the dect wireless protocols. In *Proc. 1995 IEEE Workshop on Factory Communication Systems, WFCS'95*, Leysin, Switzerland, 1995.
- [12] Philip Morel and Alain Croisier. A wireless gateway for fi eldbus. In *Proc. Sixth International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 95)*, 1995.
- [13] Philip Morel and Jean-Dominique Decotignie. Integration of wireless mobile nodes in map/mms. In *Proc. 13th IFAC Workshop on Distributed Computer Control Systems DCCS95*, 1995.
- [14] The Editors of IEEE 802.11. *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, November 1997.
- [15] The Editors of IEEE 802.11. *IEEE Standard for Information Technology - Telecommunications and information exchange between systems - Local and Metropolitan networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher speed Physical Layer (PHY) extension in the 2.4 Ghz band*, 1999.
- [16] The Editors of IEEE 802.11. *IEEE Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: High Speed Physical Layer in the 5 GHz band*, 1999.
- [17] PROFIBUS Nutzerorganisation e.V., PROFIBUS Nutzerorganisation e.V., Haid-und-Neu-Str. 7, Karlsruhe, Germany. *Implementation Guide to DIN 19245 Part 1*, August 1994.
- [18] Hideaki Takagi. *Analysis of Polling Systems*. MIT Press, Cambridge, Massachusetts, 1986.
- [19] Hideaki Takagi. Queueing analysis of polling models: an update. In Hideaki Takagi, editor, *Stochastic Analysis of Computer and Communication Systems*, pages 267–318. Elsevier, Amsterdam, 1990.
- [20] Union Technique de l'Electricité. *General Purpose Field Communication System, EN 50170, Volume 2: PROFIBUS*, 1996.
- [21] Andreas Willig. *Investigations on MAC and Link Layer for a wireless PROFIBUS over IEEE 802.11*. PhD dissertation, Technical University Berlin, Department of Electrical Engineering and Computer Science, December 2001. submitted.
- [22] Andreas Willig, Martin Kubisch, Christian Hoene, and Adam Wolisz. Measurements of a Wireless Link in an Industrial Environment using an IEEE 802.11-Compliant Physical Layer. *IEEE Transactions on Industrial Electronics*, 2001. accepted for publication.
- [23] Andreas Willig and Adam Wolisz. Ring Stability of the PROFIBUS Token Passing Protocol over Error Prone Links. *IEEE Transactions on Industrial Electronics*, 48(5):1025–1033, October 2001.
- [24] Zhensheng Zhang and Anthony S. Acampora. Performance of a modified polling strategy for broadband wireless lans in a harsh fading environment. *Telecommunication Systems*, 1:279–294, 1993.