Wireless LAN Technology in Factory and Industrial Automation

Andreas Willig, *Member, IEEE,* Adam Wolisz, *Senior Member, IEEE*

Abstract

Fieldbus systems are a mature technology designed for applications in industrial and factory automation, where (hard) real-time requirements have to be fulfilled under harsh environmental conditions. The idea of using wireless technology on the factory floor is appealing, since automation components can be mobile, and furthermore the need for (breakable) cabling is reduced. In this article we survey the requirements, problems and existing approaches for creating wireless fieldbus systems. One of the most fundamental problems comes from the fact that the wireless transmission characteristics are much different from those of other media types, leading, amongst others, to comparably high and time-varying error rates. This poses a significant challenge for fulfilling the hard real-time and reliability requirements of industrial applications. We discuss some mechanisms to improve a stations ability to deliver important packets timely and with the required reliability over wireless links, and we also provide a conceptual framework for comparison of alternative mechanisms. The second important problem area is the creation of so-called *hybrid systems*, i.e. systems where wired and wireless fieldbus stations run in the same network. We survey for different popular fieldbus systems the specific problems and solutions for integrating both media types.

Index Terms

wireless LAN, fieldbus systems, hard real-time communications, hybrid systems

I. Introduction

The technology of fieldbus systems is a mature technology [105], [89], [106], [29], [87], specifically designed to provide hard real-time communication services in harsh industrial environments for supporting implementation of distributed industrial control systems. There are many fieldbus systems standardized and commercially available, e.g. PROFIBUS [130], FIP/WorldFIP [131], and IEC Fieldbus [59], to name but a few. Initially, all these systems were specified for use with wired media, e.g. copper cables or fiber cables, and their protocol design took advantage of the low error rates achievable with these media types. With the advent of commercially available wireless transmission technology, however, the idea to replace cables with wireless media and to obtain a wireless fieldbus emerged [28], [140], [135], first publications on this topic date back to 1988 [82]. This interest grounds in the potential benefits of wireless transmission technologies:

- Reduced cabling: installing cables can be costly and time-consuming.
- Mobility: mobile subsystems can be coupled to stationary systems, increasing plant flexibility and reducing the danger of production stops due to cable breaks.

- In vibrating environments or in case of moving robots cables become a vulnerable component of the system.
- Flexibility: for temporary use of diagnosis and programming stations no extra points of attachment and no extra cabling have to be provided.
- Sometimes wireless communication is the only option, e.g. in environments where cables would have to go through chemically aggressive environments, or where people can be harmed when trying to install wires.

Applications like distributed control and distributed robotics (for example [115]) can benefit significantly from wireless communication technologies [69], [63]. However, despite these benefits wireless technology is by no means as widely accepted in manufacturing plants as it is in office and home environments, where wireless local area networks (WLAN's) have become a commodity. This can be attributed to some fundamental problems associated with WLAN technology.

In this paper we look into the issues, problems and solutions involved in adopting wireless transmission technology for fieldbus systems. One major problem are transmission errors on wireless channels, with time-variable and sometimes quite high error rates. As fieldbus systems are used for their ability to (almost) guarantee the delivery of important data packets within well-known and fixed time-bounds, the presence of non-negligible error rates has the potential to compromise the expected quality and to harm the suitability of wireless technology for certain industrial applications. We demonstrate by example of the PROFIBUS fieldbus that existing protocols have unexpected problems regarding timeliness and reliability when subjected to the error conditions of a wireless link. Somewhat overstated: so far fieldbus systems are designed with the assumption that channel errors are a rare event, but with wireless technology this assumption is no longer true. Wireless is different, and careful protocol engineering is needed to create appropriate protocols usable in industrial applications. We discuss appropriate performance metrics for evaluating the timeliness and reliability properties of lower layer protocols, and we also survey some of the existing techniques which can be used to achieve these goals. In our opinion the central question in wireless fieldbus research is: can we achieve the same service level in terms of timeliness and reliability when using wireless media? What is the achievable level useful for? How can we compare different approaches in this respect?

There are further problems with running the existing fieldbus protocols unmodified over wireless media. Some of the protocols used in wired fieldbus systems rely on features which are simply not available on wireless media, for example the ability to send and receive simultaneously on the same channel. Taking these things together, existing protocols should be re-engineered or new protocols should be designed for wireless fieldbus systems.

Because of the already discussed problems and benefits, wireless fieldbus systems should be used when their advantages outweigh the problems, otherwise wired systems should be the solution of choice. Thus, we will frequently face the challenge of integrating wired and wireless stations into an efficient and operable joint installation. In such a *hybrid system* both wired and wireless stations run in the same network and can communicate with each other. We discuss for certain popular fieldbus systems some of the problems and solution approaches for this integration. If, as suggested above, specifically tailored protocols are used on the wireless side, the wireless and wired protocols need to be interoperable. In addition, the wired stations should not need to change their protocol stack, even when communicating with wireless stations. This *transparency* is one of the prime requirements for hybrid systems and poses interesting technical challenges regarding proper integration.

The field of wireless and hybrid fieldbus systems has a comparably sparse footprint in the literature, and at the time of writing it can be considered an open field. We survey existing literature, indicate some open research issues, and discuss some of our own work in this field.

The paper is structured as follows: after giving a brief introduction into the major characteristics of fieldbus systems in Section II, we discuss in the following Section III the most important requirements and problem areas for wireless fieldbus systems, ranging from mobility support to security problems. In Section IV we look in some more detail into the problems and issues involved in transmitting (hard) real-time data over wireless / error-prone channels, including the different sources of time losses, appropriate performance metrics, and mechanisms to increase transmission reliability and timeliness. The discussions are generic, most considerations apply to several systems. After this, in Section V we start the discussion of *hybrid systems*, i.e. those systems where wireless stations are added to an existing and productive wired fieldbus system. We present existing research efforts for some important fieldbus systems, including PROFIBUS (Section VI), FIP/WorldFIP (Section VII), Controller Area Network (Section VIII) and some other systems (Section IX). In the following Section X we briefly address some further issues, namely the problem of using TCP over wireless fieldbus systems and also multimedia support. Finally, in Section XI we provide our conclusions.

II. MAJOR CHARACTERISTICS OF FIELDBUS SYSTEMS

Fieldbus systems are a special class of local area networks specifically designed to deliver (hard) real-time services under harsh environmental conditions [105], [89], [106], [29]. Their typical applications are in distributed control systems, industrial and factory automation, and also in process automation. Most systems cover a limited geographical area of at most a few hundred meters diameter and offer low to medium bitrates, for instance the PROFIBUS [130] allows up to $12 \left[\frac{\mathrm{Mb}}{\mathrm{s}}\right]$,while FIP/WorldFIP [131] supports $2.5 \left[\frac{\mathrm{Mb}}{\mathrm{s}}\right]$ and $5 \left[\frac{\mathrm{Mb}}{\mathrm{s}}\right]$ as highest bitrates.

Fieldbus systems are often used to couple sensors and actuators to a controller station, the latter executing a control loop consisting of reading sensor data, executing the control law and writing data to the actuators. Therefore, the typical traffic mix found in fieldbus systems differs from traffic in office / home environments. It is characterized by mostly small message lengths for the control traffic, the presence of synchronous (i.e. nearly periodic with some tolerable jitter) and asynchronous traffic, occurence of event showers / alarm storms, and the need for priorities to distinguish between important messages (for example alarms) and unimportant messages [52], [105]. Some systems support even isochronous traffic. We refer to either synchronous or isochronous data also as *periodic data*.

The control loops often require that the time between two sensor readings is in the range of milliseconds to tens / hundreds of milliseconds. For these periodic data some infrequent packet losses or jitter may or may not be tolerable. Fieldbus systems are also used to couple several intelligent controllers, allowing them to exchange synchronization signals, alarms, and so on. For these asynchronous data packets we also often need timely (milliseconds to tens of milliseconds) and reliable delivery. In hard real-time communications we require both timeliness and reliability, because missing a deadline may have serious consequences. This is opposed to so-called "soft real-time" communications [26], [10], where a certain amount of packet losses is tolerable. For example: in Voice over IP systems a packet loss rate of $\approx 1\%$ (depending on the voice coding / decoding method) is considered acceptable [51].

The protocol stack of almost all fieldbus systems covers only the layers one (physical), two (medium access control (MAC) and link layer) and seven (application) of the OSI reference model. A key component for the fulfillment of timing and reliability requirements is the MAC layer. In fieldbus systems deterministic MAC protocols like TDMA, token-passing, polling or priority arbitration protocols are used instead of stochastic protocols like CSMA or ALOHA [106], [29]. An important requirement for these protocols is the support for priorities, to distinguish important messages like alarms from unimportant ones like file transfers. To improve the responsiveness of the system and to avoid blocking of the medium, most fieldbus systems have small maximum packet sizes. For example, the PROFIBUS [130] allows a maximum of 247 bytes of user data in a packet.

Fieldbus systems offer different communication models. Several systems, including PROFIBUS [130] and Foundation/IEC Fieldbus [59], rely on direct communication between individually addressable stations, similar in spirit to Ethernet and other IEEE 802.x LANs. On the other hand, systems like CAN [61] and FIP/WorldFIP [131] implement a "real-time database", where the owner of a data item broadcasts it periodically or upon request from a dedicated station (example: in FIP/WorldFIP the *bus arbiter* has a table with variable identifiers, which are polled periodically) and all interested consumers copy the data item into internal buffers, from which applications fetch the data. In these systems it is often important for the user to know the "age" or "freshness status" of a consumed variable.

Fieldbus systems mediate between a controller and the physical world. The same is true for other system types like supervisory control and data acquisition (SCADA) systems, wireless telemetry systems or wireless sensor networks [2], [107], [36]. However, as opposed to fieldbus systems, these systems are not supposed to be part of control loops with tight timing requirements or to reliably exchange time-critical synchronization signals over them. We do not consider them in this paper.

III. Challenges and Requirements

In this section we briefly touch upon some of the key requirements and problem areas for wireless fieldbus systems. The discussion of real-time transmission over wireless channels is deferred to Section IV.

A. Integration of Wired and Wireless Stations / Hybrid Systems

As already stated in the introduction, there is a vast number of existing and productive fieldbus installations and for most companies it is hardly an option to replace the existing systems completely with wireless ones. Instead, wireless stations (i.e. stations with wireless transceivers) and wireless subsystems should be *integrated* into existing wired systems, thus creating *hybrid systems*. This is well to be distinguished from systems where two wired fieldbus segments are coupled with a wireless link: all stations are wired stations and merely a piece of cable is replaced with a wireless link.

The most important requirements for hybrid systems are the following:

- Transparency to wired stations: these should not need to change their protocol stack or applications.
- Portability of application layer software: almost all fieldbus systems are specified only on the layers one (physical), two (medium-access control and link layer) and seven (application) of the OSI reference model. While the lower layers are heavily influenced by the properties of the wireless medium, the interface that the link layer offers to higher layers should be the same for both wired and wireless stations to ease porting of application layer protocols and software. Optimally, not only the syntax but also the semantics of the link layer interface is preserved, including its real-time and reliability properties.

The protocols used in wireless stations can be the same as for wired stations or they can be different. In the latter case, it is required that the wireless protocols integrate smoothly with the wired ones.

Hybrid systems are discussed in more detail in Section V.

B. Mobility Support

The support for station mobility is a key feature of wireless systems. Similar to cellular systems like GSM, wireless stations can move from the range of one base station / access point to another one. The stations communication should continue without any disruptions and with maintaining the desired level of service. Ideally, even during its transition a wireless station answers to requests within few milliseconds and gets its own requests answered within the same time frame. On the other hand, the achievable service quality and the needed protocol support depend on the maximum speed of mobile stations, their mobility patterns, etc. For most of the mobile systems in a manufacturing plant (humans, guided vehicles, forklifts, moving parts of a machine) a maximum speed of $20 \left[\frac{\text{km}}{\text{h}}\right]$ can be assumed (see [52], based on a user survey). For assessing the performance of handover schemes and the suitability of a particular placement of wireless base stations / wired-to-wireless gateways, *mobility models* are needed. To the best of our knowledge there are no published mobility models specific for industrial applications. However, it is conceivable that the popular random waypoint model [13], [65] can be used for modeling the behaviour of guided vehicles, mobile subsystems etc.

Mobility may introduce significant dynamics into fieldbus networks, in the sense that inclusion and exclusion of stations is not a (very) rare event anymore. Many fieldbus systems and applications are designed with the assumption that a particular installation is set up once and not altered (often) after this. For example, in the FIP/WorldFIP protocol [131] the polling table of the bus arbiter has to be changed to accommodate a new station. Furthermore, the available address space can be rather small, for example in the PROFIBUS there are at most 127 distinct station addresses available [130]. As a consequence, in fieldbus systems the address management is often done manually and there are no protocols like ARP (address resolution protocol) and DHCP (dynamic host configuration protocol), giving automated support for dynamic address management. As an example, consider a large manufacturing plant with several PROFIBUS networks, each having a wireless extension. Assume that each member of the plants service staff has a laptop or a PDA with a wireless PROFIBUS adapter, needing a PROFIBUS MAC address. Which addresses are assigned to the PDA's? Without address management protocols one could keep the same portion of the address space free in each PROFIBUS LAN and configure the PDA addresses either statically or upon attaching to a network. However, in the latter case the user has to negotiate with other dynamic users already active in the same network to avoid assigning the same address twice.

C. Security Aspects

Security played no important role in the initial design of fieldbus standards, because:

- Eavesdropping, insertion of malicious packets and producing destructive interference (jamming) would require to physically tap a cable, which can be prevented by simple administrative measures.
- Fieldbus systems were supposed to be used exclusively in the manufacturing plant with no direct connection to other networks like the Internet.[1] Therefore, the nowadays common threats like hackers, denial-of-service attacks, viruses, etc. were not anticipated.

Both reasons are not valid anymore. Todays automation networks tend to be more and more integrated with other networks, for example to allow cost-effective remote monitoring and maintenance of machine plants. There are many techniques to protect a network against attackers from outside, for example firewalls [117]. For this source of threats the (wireless) fieldbus system needs no own security mechanisms, since firewalls are typically placed at the fringe of factory Intranets, some hops away from the fieldbus. But when using wireless media an attacker which is close enough to the network (say, on a company's parking lot) can do the following things:

- eavesdropping: an attacker might record process data and commands, for instance to find out details about a companies production process. Encryption could be used to prevent this (for example the wired-equivalent privacy mechanism of IEEE 802.11 [126]), but since the set of messages transmitted from a sensor / towards an actuator tends to have low entropy (example: most of the packets sent to a valve will be commands like "switch valve on" or "switch valve off"), cryptanalysis is comparably easy and may render the extra overhead for encryption useless if no additional measures are taken.[19, Sec. 1.2].
- jamming: an attacker might generate noise and prevent any useful transmission, this way harming reliable and timely data transfers (denial of service). One way to prevent jamming from outside the manufacturing plant is to weave metal threads into its walls to create a Faraday cage, or to use narrowband-jamming-resistant transmission schemes like spread-spectrum communications [44], [93]. Further mechanisms have to be developed.
- injecting packets: an attacker might generate false sensor data or malicious command packets for actors, send management packets to include or exclude stations from a network etc. To prevent this, mechanisms for ensuring authentication ("who sent this message?") and message integrity ("is this the message originally sent?") are

[1] Most fieldbus standards were developed between the mid of the eighties and the early nineties of the last century, where the Internet played no important role anyway.

needed [18], [117] to create mutual trust relationships between mobile stations and access-points / wired-to-wireless gateways. Such mechanisms are often implemented using shared secrets and public key cryptography, calling in turn for proper key distribution protocols. To avoid replay attacks proper sequence numbers / session keys have to be introduced into these protocols. In the case of hybrid systems, these protocols have to take into account that the wired stations do not run any security-related protocols and cannot participate.

In [19, Sec. 1.2] the challenges for implementing security mechanisms for wireless sensor networks are summarized, which largely carry over to the case of wireless sensors / actors in fieldbus systems:

- substantial overhead: ensuring authentication and message integrity for each message requires *message integrity check* (MIC) fields in each message. For the mechanisms to be effective, this field should have a reasonable minimal length, for example 16 bytes. However, since in most fieldbus systems the maximum allowable frame size is small (and many packets have only a few bytes anyway), the MIC fields account for significant fraction of overhead.
- implementation complexity: to be cost-effective in small sensors, the cryptographic algorithms cannot rely on the presence of special-purpose cryptographic processors, but instead have to be implemented on small microcontrollers running in the sensor. This limits algorithm complexity.
- key distribution: this task introduces significant administrative overhead.

D. Co-Existence

Even if jamming from the outside can be prevented, there is the problem of interference from devices like microwave ovens or from co-located wireless systems working in the same frequency band. As an example, IEEE 802.11b [104] and Bluetooth [17], [47] both utilize the 2.4 [GHz] industrial, scientific and medical (ISM) band and the issue of co-existence arises [27], [79], [57].

E. Energy Supply / Low Power Operation

Some fieldbus systems allow to provide a stations operating power over the cable. For wireless-only stations the question of energy supply can be either solved by wireless energy transmission [53], [31], energy-scavenging methods [114] or by using batteries. For battery-driven stations energy is a scarce resource and should be used economically. This is likely to remain in the near future, since Moore's law does not hold for batteries. From the context of wireless sensor networks [2], [107], [36], [19], [94] it is well-known that computation is much cheaper than communication, energetically-speaking, and significant energy savings can be achieved by proper organization of the protocol stack. On the other hand, energy was not a concern in wired fieldbus systems and the protocols were not designed for this specific target. It is an interesting and, to the best of our knowledge yet unexplored question, if and how energy-saving mechanisms can be integrated with mechanisms supporting reliable communication under real-time constraints. The issue of energy-conservation is discussed in a broader context in a number of publications [35], [122], [45].

For fieldbus systems we sometimes have the fortunate situation that the lifetime of a small wireless sensor monitoring a continuous and slowly varying physical processes and being periodically polled by a wired (and energy-ignorant) master station can be significantly lengthened by introducing an *energy proxy*. This proxy node answers the queries of the wired master on behalf of the wireless sensor from an internal buffer. On the other hand, the proxy updates its buffer contents from polling the wireless sensor at a much slower rate. In addition, the proxy might preprocess data destined to the wireless station in order to aggregate it or reduce its size [40].

F. Electromagnetic Compatibility

The wireless transceivers have to meet electromagnetic compatibility (EMC) requirements. They do not only have to properly restrict their radiated power and frequency range, on the other hand the transceivers are also exposed to radiation / electromagnetic fields from other transceivers, motors, high voltage electrical discharges and so on. This may pose a serious problem if off-the-shelf wireless transceivers are used (e.g. commercial IEEE 802.11 hardware), which are designed for office environments, but not for harsh industrial environments with lots of "electromagnetical dirt". Not only the shielding might be inappropriate, but these components also have to deal with more extreme humidities, temperature ranges, vibrations, etc. To the best of our knowledge, at the time of writing there are no commercially available transceivers specifically for industrial applications, so off-the-shelf components are the only option, if one does not want to develop own radio hardware. In our opinion this situation is unlikely to change, since the market for wireless industrial systems seems not to have sufficient volume to make costly development of dedicated wireless transceivers profitable.

IV. Real-Time Transmission over Wireless / Error-Prone Channels

Fieldbus systems are used because their protocols allow to transmit asynchronous and periodic data reliably / acknowledged within fixed time-bounds, this way providing hard real-time services. In industrial control applications missing a deadline is critical, whereas in "soft real-time" applications like voice or video a given percentage of deadline violations is acceptable. For soft real-time data a data packet is transmitted timely, but not much effort is spent to ensure its reliable delivery, and thus often no retransmission schemes are used. On the other hand, in hard real-time applications quite an effort is made to assure sufficient amount and modes of retransmissions, in order to achieve delivery reliability. Obviously, those retransmissions have to take place in a timely fashion. Put briefly: for soft real-time data the timing of the *first* (and single) copy of a packet is important, while in industrial contexts we care about the *last* (and hopefully successful) packet.

Therefore, an important challenge for wireless industrial LANs is to find transmission schemes and protocols that give optimum performance in terms of real-time and reliability despite channel errors and certain other specific properties of wireless channels. Since specifically the lower layers are key for implementing the real-time capabilities, it makes sense to concentrate the efforts here. The lower layer protocols should be specifically tailored for wireless media and thus they may well be different from the protocols used in wired industrial communication systems.

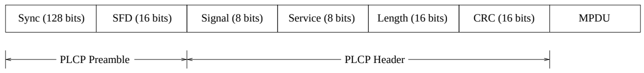

Fig. 1. Format of an IEEE 802.11/802.11b physical layer frame

It is important to give a more precise meaning to the notion of "optimum performance in terms of real-time and reliability", which we henceforth refer to as *real-time performance*. We propose a set of real-time performance metrics, which take both timeliness and reliability into account and which also reflect the error properties of wireless links. Such a set of performance metrics along with appropriate channel error and load models allows to compare different mechanisms with respect to their real-time performance.

In this section we first discuss some of the reasons by which wireless transmission can affect real-time performance, with channel errors being the most important one (Section IV-A). After this we introduce in Section IV-B the real-time performance metrics and discuss the critical role of channel error models for assessing the real-time performance by virtue of a fieldbus-specific example. In Section IV-C we present several countermeasures against channel errors and discuss their real-time performance, where results are available. In Section IV-G we summarize the discussion by pointing out areas of future research in the area of transmission of hard real-time data over wireless channels.

A. Sources of Reliability- and Time-Losses

There are several potential sources of time-loss in wireless industrial LANs. First, if the wireless stations are mobile, handovers may interrupt ongoing transmissions. Second, in the case of hybrid fieldbus networks enabling wired stations to poll wireless stations or vice versa, packets passing the wired-wireless boundary have to be forwarded through a bridge-like device. Even if fast cut-through forwarding is used, the possibly different transmission speeds of the two media and the presence of physical layer preambles of different lengths lead to a forwarding delay, which depends on the relative speed of the media [70].

1) Channel Errors: A significant source of time-losses are channel errors. It is well known that transmission on wireless channels suffers from phenomena like fast / multipath fading, intersymbol interference (both phenomena are caused by reflection and diffraction of waveforms, creating multiple copies of the same signal with different relative delays), co- and adjacent channel interference from co-located wireless communication systems, thermal and man-made noise, path loss and attenuation [111], [24], [101]. In industrial environments, significant noise as well as distortion of transceiver electronics can also be created by strong motors, static frequency changers, electrical discharge devices, etc. Measurements of some key wireless channel characteristics in industrial environments [52], [71], [109], [110], [112], [49] have shown that the delay spread can reach values larger than 200 ns (Reference [111, Table 5.1] cites other measurements in indoor environments giving similar values), and therefore modulation schemes with symbol rates beyond $1\ \left[\frac{\text{Mbaud}}{\text{s}}\right]$ are subject to severe intersymbol interference.

The distortion of waveforms translates into bit errors and packet losses. This distinction can be best explained with reference to a specific format of a wireless physical layer frame, for example the physical layer protocol data unit (PPDU) of the IEEE 802.11 / 802.11b wireless LAN standard [126], [104], which: a) has a rather typical frame format, and b) due to its commercial availability is an interesting physical layer technology for wireless fieldbus systems. The frame format is shown in Figure 1. The PPDU is subdivided into the physical layer convergence protocol (PLCP) preamble, the PLCP header and the PPDU data part, carrying a medium access control (MAC) PDU (MPDU). The PLCP preamble is a constant bit pattern and useful for equalization and to allow the receiver to acquire bit- and frame synchronization. The PLCP header describes amongst others the length and the modulation scheme used in the data part; in addition, the header is protected by its own checksum field. Bit errors occur solely in the PPDU data part, while packet losses occur if either the receiver fails to acquire bit-/frame-synchronization or the PLCP header checksum is wrong, causing the receiver to drop the entire packet. In this particular framing scheme, applying forward error correction (FEC) [84], [92] may correct bit errors in the MPDU, but the PLCP header is not covered and therefore packet losses cannot be prevented. Measurements with an IEEE 802.11-compliant radio transceiver taken in an industrial non line-of-sight (NLOS) environment [143] have shown that indeed both types of errors occur, at sometimes impressive rates. As an example we show in Figure 2 for a particular measurement campaign from [143] the obtained packet loss rates per trace. A trace consisted of 20.000 packets, and the particular measurement campaign consisted of 180 traces (spanning approximately four and half hours). All packets in all traces were transmitted using the same modulation scheme ($2\ \left[\frac{\mathrm{Mb}}{\mathrm{s}}\right]$ QPSK) and packet size. In Figure 3 we show for the same measurement campaign the bit error rates per trace. While it can be argued that these results are particular for the specific chipset and environment, they show some trends which have also been observed in other wireless measurement campaigns [42], [32], [103], [30]:

- Bit errors and packet losses are “bursty”, i.e. they occur in clusters with error-free periods (“runs”) between the clusters. The empirical distributions of the cluster and run lengths often have a large coefficient of variation or even seem to be heavy-tailed [72].
- The bit error rates depend on the modulation scheme, typically schemes with higher bit rates exhibit higher error rates. For QPSK bit error rates in the range of $10^{-3} \ldots 10^{-6}$ were observed. Hence, the wireless channel is much worse than wired channels. Furthermore, the bit error rate can vary over several orders of magnitude within minutes.

2) Channel Error Models: For analytical or simulation-based investigations of the reliability and timeliness behaviour of protocols appropriate channel error models are needed. The choice of models is critical, we show in Section IV-B for the example of the PROFIBUS that the achievable real-time performance depends strongly on it. To reflect the full complexity of the wireless channel, between each pair of stations a separate wireless channel model should be created, with each channel model having two cascaded sub-models: first, the *packet-loss submodel* decides whether a packet transmitted over this channel is lost, and in case the packet is not lost, the *bit-error submodel* decides about the number (and possibly position) of bit errors. If all channels follow the same stochastic

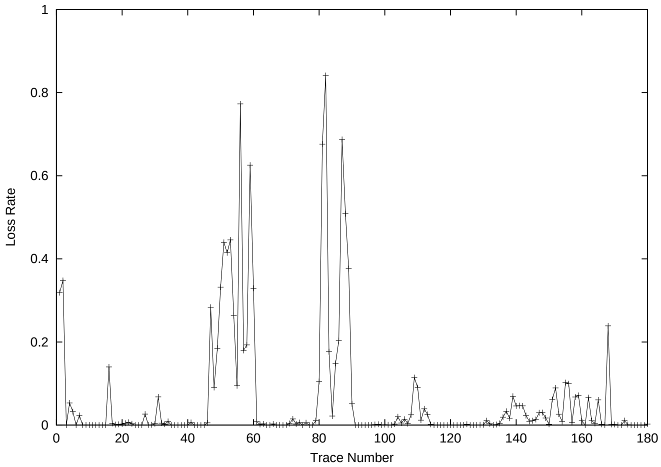

Fig. 2. Rates of lost packets for **longterm1** measurement (> 4 hours)

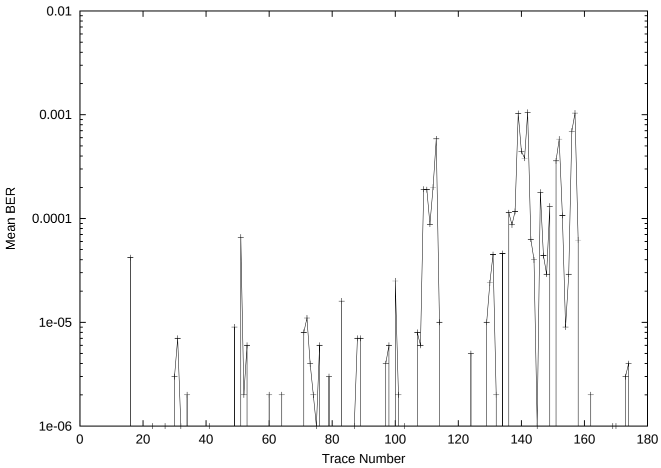

Fig. 3. Bit error rates per trace for **longterm1** measurement (> 4 hours)

model, we call the overall channel model *homogeneous*, otherwise it is called *heterogeneous*. In both cases the channels may be stochastically independent or correlated.

A single channel sub-model is typically represented by a simple stochastic process. Common choices are the *independent model* which models packet-losses / bit-errors as iid with some probability p (this is also known as a *binary symmetric channel* (BSC)), or the popular *Gilbert-Elliot model* (GE model) [43], [34], which models a channel as alternating between a "good" state and a "bad" state, such that within both states errors occur according to a BSC with different error probabilities p_g and p_b ($p_b \gg p_g$) in the good and bad state, respectively. Furthermore, the channel state is varied according to a two-state Markov chain.[2] The GE model is capable of expressing bursty error patterns, while still being analytically tractable. However, this model puts strict limitations on the distribution of the error burst lengths and error-free burst lengths by requiring them to have an exponential / geometric distribution. More complex models are needed to approximate other distributions [128].

B. Real-Time Performance Metrics for Wireless Links

If one wants to compare the "real-time performance" of different fieldbus protocols over wireless links, appropriate performance metrics are needed. The example results presented in the previous section have shown that packet loss rates and bit error rates can be high enough to easily break any *deterministic* guarantee on timely delivery, therefore we use *stochastic* metrics [139]. The point of reference for these performance metrics is the interface between the link layer and upper layers. The upper layers hand over *packets* or *requests* to transmit, while the link layer generates positive or negative *confirmations* as soon as the outcome of the packet transmission is known. The outcome is negative ("failure") if the transmitter exhausts the maximum number of trials without receiving an acknowledgement, otherwise the outcome is positive ("success"). The metrics have some similarity to the notion of predictability [121] or responsivity [90] in real-time systems. They are defined for the case of having two packet priorities (important, unimportant) and focus entirely on the behaviour of the high priority / important packets, regarding unimportant packets as "background load"; however, an extension to the case of more priorities is straightforward. The goal is to transmit the important packets reliably (i.e. acknowledged). The *confirmation delay* $D_C(i,k)$ for the k-th high-priority packet issued by station i is defined as the time duration between the arrival of the packet to station i (in a situation where there is no other pending high priority packet at i) and the time instant where the confirmation primitive is generated by i's link layer, i.e. the packets fate is known. For certain load scenarios it can be assumed that for a fixed station i the random variables $D_C(i,k)$ ($k \in \mathbb{N}$) are independent and identically distributed (iid) and have a distribution $F_{D_C(i)}(x) = \Pr[D_C(i) \leq x]$. Let $x_{99}(i)$ denote the 99% quantile of $D_C(i)$, i.e. $x_{99}(i) = F^{-1}_{D_C(i)}(0.99)$.[3] The first important performance metric is the *overall confirmation delay* $\widetilde{D_C}$, which for a number N of stations is given by:

$$\widetilde{D_C} := \max\{x_{99}(i) : i \in \{1, \ldots, N\}\}$$

i.e. the maximum of the 99% $D_C(i)$ quantiles over all stations. Clearly, the lower the $\widetilde{D_C}$ value the better. As a side-metric, for two protocols with approximately the same $\widetilde{D_C}$ performance it can be evaluated, how much of the bandwidth is still available for low priority traffic. This is captured by the *remaining bandwidth* B_L metric. A second side-metric depends on the configured maximum number of retransmissions: the *negative confirmation rate* reports the fraction of finally unacknowledged requests.

[2] An extension to N states depending solely on simple physical parameters of the channel can be found in [133].

[3] The choice of the 99% percentile is somewhat arbitrary, a 99.9% percentile would do as well. However, in practice this metric will mostly be evaluated by simulations, and choosing a higher precision would increase simulation times.

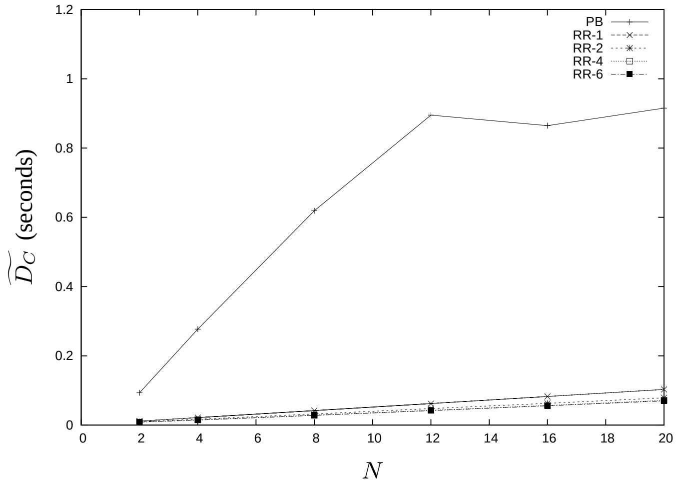

Fig. 4. Overall confirmation delays $\widetilde{D_C}$ for rr-k-protocols and the PROFIBUS protocol vs. number of WT's N for 10% background load and the Gilbert-Elliot error model.

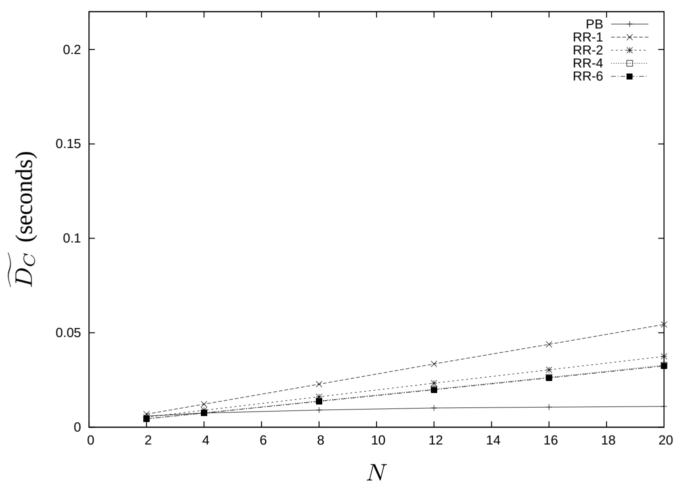

Fig. 5. Overall confirmation delays $\widetilde{D_C}$ for rr-k-protocols and the PROFIBUS protocol vs. number of WT's N for 10% background load and the independent error model.

The $\widetilde{D_C}$ value depends on the number N of stations. Conversely, if we fix a maximum overall confirmation delay δ_{max} we might ask for the *real-time capacity*, i.e. the maximum number N of stations such that $\widetilde{D_C} \leq \delta_{max}$.

While the overall confirmation delay prescribes a percentile and results in an achievable deadline value, the *failure probability* fixes a deadline value and measures which fraction of the important requests fails to meet this deadline.

It is exactly the small relaxation from a deterministic delay metric to 99% quantiles which accounts for the variability and error behavior of the wireless link, while simultaneously expressing hard delay requirements.

To illustrate these metrics and to provide an example that shows that simply re-using the existing fieldbus protocols on top of a wireless medium is not optimal, we quote some results from [139]. The real-time performance of the PROFIBUS token-passing protocol is compared to different polling protocols (k-limited round-robin, denoted as rr-k: upon reception of a polling frame a station may initiate at most k frame exchanges, including acknowledgements). Specifically, in Figures 4 and 5 we show the overall confirmation delay $\widetilde{D_C}$ for the PROFIBUS protocol and the round-robin protocols rr-1, rr-2, rr-4 and rr-6 for the independent and the GE error models, respectively. It can be seen that for bursty errors the polling-protocols have significantly better $\widetilde{D_C}$ values, while for independent errors the PROFIBUS protocol is better than the polling protocols. As will be explained in Section IV-D, the poor performance of the PROFIBUS protocol under bursty errors is caused by problems with the maintenance of the logical token-passing ring [144]; similar problems are also reported for the IEEE 802.4 token-bus protocol [67].

C. Countermeasures and their Real-Time Performance

In the following sections we present methods which can be used to combat or hide bit errors and packet losses on wireless channels. In general, this has been a topic of intensive research and an immense body of literature exists. To be relevant, we highlight those mechanisms which focus on increasing reliability and are also deadline-aware, instead of optimizing performance metrics like throughput or energy consumption. Where available, we include results on real-time performance metrics.

Most of our discussion concentrates on the MAC and link layer / error control, for the following reasons: a) these layers are in general of prime importance for fulfillment of time and reliability guarantees; and b) any progress made in these layers relaxes the requirements for the underlying physical layer and commercially available technologies (which are not designed for industrial applications in the first place!) can be used. An evaluation of some commercial physical layer technologies (including Bluetooth, DECT, IEEE 802.11 and HIPERLAN/2) for the purpose of wireless fieldbus systems has been carried in [71], and [16] provides a more specific discussion regarding Bluetooth.

D. MAC Protocols

To guarantee upper bounds on medium access delays, in most fieldbus systems deterministic techniques like time division multiple access (TDMA), token-passing, polling or CSMA with priority arbitration have been used ([106], [29], [73], [108] with focus on fieldbus and time-constrained traffic, while [46], [100], [3] provide a general survey on wireless MAC protocols). For example, the PROFIBUS fieldbus [130] and the IEEE 802.4 token-bus [60] use a token-passing scheme to pass the right to initiate transmissions, the token-owner is allowed to poll other stations for a certain amount of time; FIP/WorldFIP [131] relies on polling, and the controller area network (CAN) autobus [61] uses CSMA with priority arbitration. Some TDMA schemes have been investigated in a wireless ATM context for their deadline-miss probability (DMP) and their capacity for a given DMP target without considering channel errors / retransmissions [116], however, in the following we focus on token-passing and polling schemes, as they are more important for today's fieldbus systems.

1) Token-Passing Protocols: The token-passing protocols of IEEE 802.4 and PROFIBUS use a broadcast medium and construct a *logical token-passing ring*, or *logical ring* for short. The right to initiate data transfers is tied to a small control packet, the so-called *token frame*, which is passed along the logical ring. Upon reception of a token, a station calculates its *token holding time* and may use this time to exchange data packets with other stations. Besides token-passing additional procedures are needed for ring maintenance: inclusion and exclusion of stations, repair of lost tokens, and so on. For some of these procedures control frames need to be exchanged. There are two major problems with token-passing on wireless links. The first one is the handling of topologies where not all stations hear each other (partially meshed topologies). To deal with this problem, methods for organizing the logical ring in accordance with the (time-variable) topology would be needed (for example: [91]). None of todays fieldbus systems using token-passing specifies such a mechanism.

The second problem has to do with channel errors. It has been shown in [144], [136] for the PROFIBUS and in [67] for IEEE 802.4 token-bus that these protocols have problems with channel types exhibiting comparably high bit error rates ($\geq 10^{-4}$) and bursty errors, like wireless channels. Especially the token-passing process is vulnerable: when a station x passes the token to its successor in the logical ring y, it is x's responsibility to check that y has received the token and continues with transmissions, otherwise x immediately starts another trial to pass the token. The maximum number of trials is bounded, typically to a small value (the PROFIBUS allows a maximum of three trials). For the case of a bursty channel between x and y there is some chance that all consecutive trials fail. Then x comes to the conclusion that y is lost from the ring and tries to continue with y's logical successor z. Unfortunately, y can continue to send packets only after it has been re-included by x into the logical ring, which, however, can take quite some time. It has already been demonstrated in Section IV-B that the PROFIBUS achieves rather poor $\widetilde{D_C}$ performance for bursty errors, and the results presented in [139] substantiate that this is caused by losing stations from the ring due to repeated token losses. As is demonstrated in [136], a packet loss rate of $\approx$ 7% (independent packet losses) suffices to let the logical token-passing ring (when running over a wireless medium) being incomplete for more than 50% of the time, even at a configuration which tries to re-include stations fast. It should be noted that the PROFIBUS protocol used in these investigations already included a modification called “fast re-inclusion”, which tries to re-include a lost station as soon as possible without waiting for the standard PROFIBUS ring inclusion mechanism.

2) Polling-based Protocols: Token-passing protocols have the problem that membership in the logical ring is tied to frequent and proper reception of the token, and repeated token losses on bursty channels cause loss of a stations membership status. Polling-based protocols can be designed such that this problem is removed: a station needs to associate once with some base station / access point to enter its polling table, but afterwards the protocol does not necessitate loss of membership status (and need for re-inclusion), if a station misses some polling frames. This way, polling-based protocols “merely” have to deal with transmission errors and not with frequent re-inclusion of a lost station. Specifically, [139] shows that for bursty errors already simple k-limited round robin protocols outperform the PROFIBUS token-passing protocol significantly.

Some additional mechanisms are specified, which are tightly integrated with the k-limited round-robin MAC, and it is shown that these mechanisms can improve the real-time performance ($\widetilde{D_C}$ performance) of k-limited round-robin under a variety of load models and channel error models. The *simple data relaying* (SDR) mechanism applies if a polled station x needs to retransmit a data packet to another station y. Since in real-time systems we have hard timing-constraints, the retransmission should take place as soon as possible, but for a bursty channel $C(x, y)$ between x and y there is some chance that an immediate retransmission will fail, too. Therefore, another channel is used: if the base station (BS) a has captured x's first packet towards y and registers lack of an acknowledgement frame, it transmits the captured data packet over the channel $C(a, y)$ to y (causing x to postpone its own retransmission) and after this forwards y's acknowledgement frame over the channel $C(a, x)$ to the initiator station x. This approach takes advantage of two circumstances: a) the stations x, y and a have different locations; and b) if channel errors are dominated by multipath fading, two receivers s and t at the same distance to a transmitter but having a mutual distance of at least 40% of the wavelength see largely uncorrelated signals [111, Chap. 5]. Therefore there is a reasonable chance that the channel $C(a, y)$ between the BS and the target station y is in good state while the channel $C(x, y)$ between x and y is in bad state.

The *simple poll relaying* (SPR) mechanism targets the problem that the channel $C(a, x)$ between the BS a and some station x is currently bad when a attempts to transmit polling packets to x. If a experiences no reaction on its poll packets for some time, another station y is asked to poll x on behalf of a (and over a different spatial channel) and to inform a about the outcome. Both the SDR and SPR schemes provide a kind of *cooperative diversity* [78]. It is demonstrated in [139] that these additions can improve the real-time performance of k-limited round-robin significantly. As an example, we show in Figure 6 that k-limited round-robin enhanced with SPR (rr-k+SPR) outperforms plain rr-k in terms of $\widetilde{D_C}$ for a scenario with 50% low priority background load and a complex channel error model between stations. In [142] another polling-based protocol, the *adaptive-intervals* protocol has also been shown to outperform the PROFIBUS protocol significantly in terms of $\widetilde{D_C}$ performance for a Gilbert-Elliot channel. This protocol adds a group testing feature [8], [22] with load-dependent group sizes to the basic polling algorithm.

Another study of polling-based protocols over wireless links is presented in [38], where it is proposed to let the base station preempt a station enjoying exhaustive or gated service in case it experiences a deep fade (a Gilbert-Elliot model is used) and performs retransmissions, and to service other stations in the meantime. Results on throughput and delay are presented, showing improved performance of the proposed protocol over a TDMA solution. However, deadlines are not taken into account. Further simulation-based investigations of polling systems over bursty channels described by a Gilbert-Elliot model are available in [148] and [54], where results concerning cycle-time properties and throughput / mean-delays are presented. For the simpler case of a binary symmetric channel analytical results for the mean message response time are presented in [124], [125]. In [147] polling- and priority mechanisms have been considered for use in networked control systems (i.e. control systems with a network in the loop), and the achievable control performance of different schemes for scheduling the real-time traffic have been evaluated. However, channel errors were not taken into account. The suitability of polling-based protocols, TDMA and random access protocols for networked control systems has also been investigated in [88].

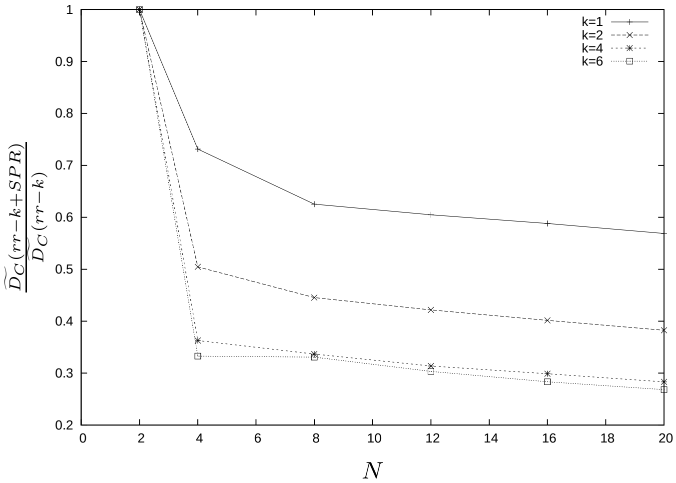

Fig. 6. Ratio of the Overall confirmation delay for the rr-k+SPR protocol and the rr-k protocol $\frac{\widetilde{D_C}(rrk+spr)}{\widetilde{D_C}(rrk)}$ vs. number of wireless stations N for 50% low priority load and complex error model

3) Priority Arbitration Protocols: In priority arbitration protocols each message is tagged with a priority value, and this value is used to resolve collisions deterministically between contending stations. When priorities are properly assigned to the different (periodic) data sources (for example according to a rate-monotonic scheduling approach [85]), this kind of protocols can satisfy deadline requirements. For instance, in the CAN protocol all stations are tightly synchronized and the priority field is at the beginning of packets. Each backlogged station transmits its priority field bit-by-bit and reads back the signal from the medium. If the medium state is the same as the transmitted bit, the station continues, otherwise the station gives up and waits for the next contention cycle. These protocols rely on the medium property to produce a logical one on the medium if all stations transmit a logical one, and to produce a logical zero on the medium if at least one station produces a logical zero. This protocol therefore requires immediate hearback from the medium, however, wireless transceivers do not allow to transmit and to listen simultaneously on the same channel. Alternative mechanisms for deterministic collision resolution have been developed in the context of wireless CAN systems [77], [74], [75], [76], sketched in Section VIII.

E. Error Control

Error control methods [86], [92], [84] are of utmost importance to ensure timely and reliable data transmission and have been a subject of research for a long time. For fieldbus systems following the real-time database paradigm, open-loop methods like open-loop FEC [15] or receiver diversity are most appropriate, since in broadcast-based systems it is hard to provide feedback to the transmitter. For systems relying on direct communication between addressable stations both open-loop and closed-loop error control methods are applicable. However, for our purposes of stochastic hard real-time communication we are specifically interested in closed-loop methods like automatic repeat request (ARQ) protocols [48] or hybrid protocols (a combination of ARQ and FEC), which allow the system to react properly on transmission failures, and to give small residual error / loss rates. Open-loop methods cannot correct all channel errors, specifically not in case of a deep channel fade.

Closed-loop error control schemes rely on retransmissions. On wireless channels with bursty errors retransmitting the same packet immediately over the same channel and to the same receiver is not the best idea, since the retransmitted packet may be hit by the same error burst as the original packet. Therefore, of particular interest for wireless industrial LANs are schemes which try to treat retransmissions in more clever and timing-aware ways, in order to keep deadlines and to avoid blocking of competing packets by retransmissions.

1) When to retransmit?: A lot of retransmission schemes retransmit the same packet over the same channel but not immediately, in the hope that the error burst hitting the first copy has ended in the meantime. Therefore, these schemes try to determine a time instant where the channel is believed to be in good state again, either by using small probing packets [149], by channel prediction based on measurements or by a-priori knowledge / an educated guess. It has been shown [14], [20], [21] that postponing retransmissions and serving packets destined to stations at different locations can significantly increase the overall throughput. However, in these schemes the postponing decision is based solely on the channel state and not on packet deadlines. In [33] a scheme is described which takes both the estimated channel state (for postponing decisions) and the packet deadline into account to select one coding scheme from a suite of available schemes. It is shown for a Gilbert-Elliot-type wireless channel that their scheme reduces the bandwidth needed to achieve a prescribed maximum failure probability as compared to a static scheme taking only the channel state into account.

2) Where and who to retransmit?: Another line of attack would be to use different channels for the packet retransmissions. These can be either different frequencies or different spatial channels. An example for the latter is discussed in [138], [141]. Consider a scenario where a central station (or wired-to-wireless gateway) transmits and receives packets from wireless stations. The central station is equipped with multiple antennas, say: K antennas, which best are located at different locations. In the uplink direction this setup provides receiver diversity, while in the downlink direction the K antennas can be used to transmit retransmission packets over spatially different (and hopefully uncorrelated) channels. For example: the first trial of a packet is sent over antenna one, the first retransmission over antenna two, the second retransmission over antenna three, and so forth. This scheme therefore belongs to the class of *transmit diversity* techniques, where multiple antennas can be used to increase transmission robustness (for example: [4] or the soft handover technique employed in UMTS [132]) or to increase throughput [41], the latter at the expense of requiring multiple antennas and significant signal processing at the receiver. In contrast, the scheme discussed in [138], [141] requires only a single antenna and comparably simple receivers. For the case of K independent Gilbert-Elliott-type channels between a wireless node and the K antennas, in [141] the failure probability with respect to a given deadline for a downlink packet is evaluated in a setting where the multiple transmit antennas and FEC coding are combined. The joint effects of antenna redundancy and FEC are illustrated in Figure 7 for a scenario with a varying number of antennas K, mean bad state lengths of $10\ [\text{ms}]$, mean good state lengths of $65\ [\text{ms}]$, a channel bit rate of $1\ \left[\frac{\text{Mb}}{\text{s}}\right]$, a packet length of $416\ [\text{b}]$ and a deadline D corresponding to ten trials. It can be seen that antenna redundancy reduces the failure probability by almost one

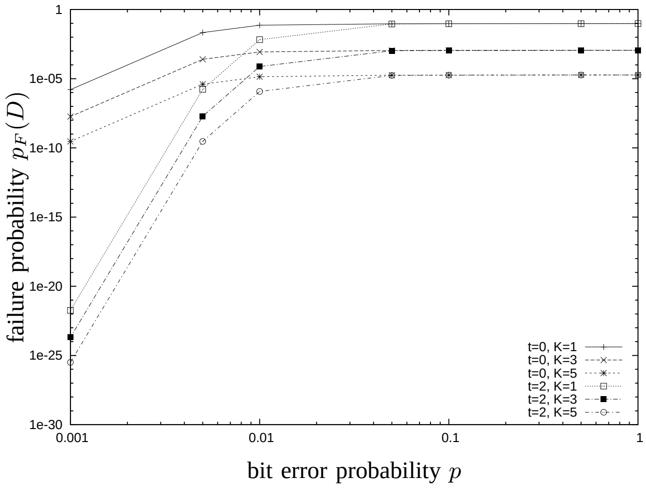

Fig. 7. Failure Probabilities vs. bit error probability p during bad state for the experiment "Efficiency of Redundancy Approaches", $t \in \{0, 2\}$

order of magnitude for each additional antenna, but also that if the bit error probability p during the bad state is smaller than some threshold ($\approx 5 \cdot 10^{-3}$ in the figure) it pays out more to invest into FEC. It is also shown that multicopy-ARQ approaches (where a single packet is transmitted several times to hopefully increase its reception probability [9]) are not helpful for the investigated Gilbert-Elliot channels, since they tend to increase the failure probability.

3) How to retransmit?: The transmitter can change the contents and format of retransmitted packets in several ways; similarly, the receiver can try to take more advantage of the information contained in the erroneous packet and the retransmissions.

On the transmitter side, one approach is to change the amount of FEC coding applied to a packet. In [129], [12], [33] the strategy to increase the coding strength (decreasing the code rate) more and more as the packet deadline comes closer ("deadline-dependent coding") has been adopted. Another way to decrease the bit error rate would be to switch to more robust modulation schemes (see [119], [120], with consideration of energy consumption) or to increase the transmit power as the deadline comes closer. In [55] a dynamic programming problem is specified, which takes a Markov chain model for the wireless channel and a Markov chain model for the data traffic as inputs (both discrete-time). In the latter, to each state of the model a deadline and a number of arriving packets is associated. The solution of this dynamic program provides a policy for controlling transmit power, modulation scheme and FEC coding for original packets and retransmissions such that the deadlines are met with a prescribed probability and the power consumption is minimized.

On the receiver side advantage can be taken of the information contained in already received erroneous packets, for example by using packet combining methods like equal-gain combining or bit-by-bit majority voting [50], [134], [68]. These are examples of type-II hybrid ARQ [86] approaches. Another way to partially recover information from received packets is to partition the packets into several chunks, with each chunk having its own checksum

[83]. The receiver can store the correctly received chunks and request only the missing ones for retransmission.

F. Further Considerations

An interesting control knob for improving the real-time performance is the choice of maximum packet size. In many fieldbus systems, this size is already low, for example the maximum packet length in PROFIBUS is 255 bytes, carrying 247 bytes of user data. It is well-known that on wireless links proper choice of packet size can increase throughput or decrease energy consumption [83], [95]. However, in wireless industrial networks also the responsivity of the network has to be taken into account: longer packets, which often belong to timing-uncritical transmissions like file transfers, are more prone to errors and consequently need to be retransmitted more often on average than short and time-critical packets, this way blocking the channel for timing-critical transmissions. Just as an example: for a constant bit error rate of $p = 0.001$ a 255 byte packet needs on average ≈ 7.7 trials to be transmitted successfully (corresponding to ≈ 160 [ms] transmission time on a $1 \left[\frac{\text{MBit}}{\text{s}}\right]$ channel), whereas a 30 bytes packet needs only ≈ 1.3 trials (corresponding to ≈ 0.3 [ms]). This favors the choice of small maximal packet sizes, even at the expense of throughput.

Sometimes it may not be possible for the lower layers to correct all channel errors. This is intolerable for asynchronous events, but for periodic data this can be different. One can simply accept occasional losses, or try to *conceal* them. For example, for slowly varying physical processes monitored by a sensor at a sufficiently high sampling rate one can replace missing samples at the receiver by an estimated value. In [56] a scheme is proposed where the receiver estimates missing values on the basis of Kalman filters, and it is demonstrated that by this technique certain signal classes need only five out of 100 samples to be able to reconstruct the signal with good quality.

G. Summary

The problem of transmitting (hard) real-time data over a wireless channel with many errors is challenging. Several mechanisms have been developed to increase transmission reliability, however, often their real-time performance is unknown. Therefore, there is a significant potential for research. Interesting topics include the development of further mechanisms and the assessment of the real-time performance of (combinations of) several of these mechanisms. In addition, the dependence of the real-time performance on the channel error characteristics is of some importance, for example if used over a BSC or over a bursty channel. Since wireless channel error characteristics can hardly be expected to be invariant, proper methods for adaptation are needed.

For comparing different mechanisms, it is useful to agree on appropriate performance metrics, load scenarios, channel error models, mobility models, etc. If energy consumption is a concern, we also need models for the energy consumption of nodes in transmit mode, receive mode, idle mode, sleep mode, and so on.

In our discussion we have not touched upon the issue of minimizing energy consumption while providing a certain level of real-time performance. To the best of our knowledge, this issue is so far unexplored.

V. Integration of Wired and Wireless Stations / Hybrid Systems

There is a huge number of existing and productive fieldbus installations and for most companies it is hardly an option to replace existing and revenue-generating systems completely by wireless ones. Instead, wireless stations and wireless subsystems have to be *integrated* into existing wired systems, such that both types of stations run within the same or within closely coupled LANs. Such systems are called *hybrid systems*. The most important requirements for these systems have already been discussed in Section III-A. In the following sections we survey the current research in creating hybrid systems for different fieldbus systems. Due to their market relevance most efforts and publications so far concentrate on PROFIBUS, FIP/WorldFIP and CAN; for other systems almost no publications seem to be available. There is some emphasis on the PROFIBUS, since for this system more publications are available than for other systems.

We use the following notions: a *wired station* is a station with a wired transceiver, a *wireless station* has a wireless transceiver, and *hybrid stations* have both. At the medium boundaries *coupling devices* have to be used, if wired and wireless stations are coupled to run within a single network. These devices can have several tasks. At minimum they are required to forward frames from one medium to the other, translating between the different framing rules used on wireless and wired media. Further translation work is needed in cases where different MAC and link layer protocols are used. Additionally, these devices may help with mobility management or security. All this work is subject to the constraint that the wired stations should not need to modify their protocol stack, and thus do not support the coupling device in solving its task.

VI. Wireless PROFIBUS

The PROFIBUS [130] is a European standard and one of the most widely used fieldbus systems. On the link layer it offers three semi-reliable and one unreliable service, each with two different priorities (low and high), and on the MAC layer a token-passing protocol is combined with polling, such that the right to initiate data transfers is passed around with the token; the current token owner is said to have the role of a *master station* and all other stations have the role of *slave stations*. The token-passing protocol includes the necessary ring maintenance procedures. On the physical layer the most common choice is a bus structure based on RS-485. There are two different types of stations: *active stations* participate in the token-passing process and might become a master station from time to time, while *passive stations* do not participate in token-passing and have the role of slaves all the time.

A. Problems of the PROFIBUS protocol

The PROFIBUS protocols are not ready for use in wireless environments. There is no mobility support, and there are no security mechanisms built into the protocol. Furthermore, the token-passing scheme is unable to cope with partially meshed topologies and has problems with real-time performance under bursty channel errors with non-negligible rate, see Section IV. These things have to been resolved in a manner compatible and integrable with the behaviour of wired stations. In addition, there is the problem of managing the limited address space, already discussed in Section III-B.

Another problem of the PROFIBUS protocol is its approach to implementing the semi-reliable link layer services. Each active station runs for each potential slave a separate instance of the alternating-bit protocol (ABP) [48]; for each packet the number of trials is bounded by a configurable number max_retry. This number is the same for low priority and high priority packets. The service for a single packet is non-preemptive, i.e. once its service started, it blocks other packets from the same station as well as other stations packets until the outcome (success, failure) is determined. To achieve high reliability for high priority packets over wireless links, a large value of max_retry may be favored, but this way low priority packets can block high priority packets for longer time. This problem is amplified by the observation that high priority control traffic tends to have small packet lengths, whereas low priority traffic is often used for file transfers, transfer of configuration data, and so on. These data types use longer packets which, however, are more likely hit by bit errors and therefore need more retransmissions than shorter packets. Therefore, a preemptive service would be appropriate for wireless links with bursty errors. To avoid duplicates at a selected responder, an active station needs to run two instances of the ABP for each responder, one instance per priority class, and possibly with different max_retry parameters.

In the following we discuss four approaches. Three of them integrate wired and wireless stations in the lower layers, while the last one can be classified as wireless application layer gateway approach.

B. Single-Logical-Ring Solution

The European Union R-Fieldbus project [52], [113] followed the goal of creating a hybrid wired/wireless PROFIBUS system supporting the real-time requirements needed in industrial control applications and having the ability to carry multimedia traffic. All wired and wireless stations run the PROFIBUS token passing protocol on top of their respective physical layer. In the single-logical-ring (SLR) solution the wired and wireless stations are integrated into a single logical ring. Thus the token is passed between wired and wireless stations without further distinction.

Wireless stations are grouped into two different types of *domains*, each domain is coupled to the wired network through a *radio repeater station*. In the first domain type, the so-called *direct link network* the belonging wireless stations can communicate with each other directly, without relaying the communications through other stations. In this kind of networks only intra-domain mobility is supported, since the wireless stations run the original PROFIBUS protocol (which has no mobility support) and the radio repeater stations just forward packets. In the second type of domains, the so-called *base station networks*, the radio repeater is more intelligent and provides support for inter-domain mobility [6], [7].

In any real-time system the achievable worst-case response times / transaction times are important. In the SLR solution it may happen that a wired master requests data from a wireless slave or vice versa. In either case the request and answer packets have to cross one or more media boundaries. In [5] the dependence of the worst case transaction durations on the number of media changes and involved media types are investigated. The different medium speeds necessitate insertion of extra idle times between subsequent packets in order to avoid congestion in the coupling devices [7]. These idle times depend on the allowable maximum frame length and therefore influence also the worst-case transaction times. In [70] the forwarding delays between wired and wireless media types are investigated more closely. Together, the idle times and the forwarding delays can increase the worst-case transaction times significantly compared to the wired-only case without media crossings. In addition, the idle times and the forwarding delays affect certain parameter settings [39], for example the so-called *slot time*, which is the time before an acknowledgement for a data frame must have been arrived. This time plays also a crucial role in detecting lost tokens. The additional delays require to set the slot time to high values, thus reducing throughput and responsiveness to token losses.

The advantages of the SLR solution are its conceptual simplicity and the fact that existing protocols can be re-used, which eases protocol engineering. However, the simulation-based investigations presented in Section IV-D raise the question how this approach behaves under significant wireless channel errors or under mobility (what happens if the token owner goes out of the range?) in terms of real-time performance.

C. Multiple Logical Ring Solution

As an alternative to the single logical ring solution adopted in the R-Fieldbus project also the approach to use multiple logical rings (MLR) was investigated [39]. The stations are grouped into so-called *domains*, where a domain consists of all stations which can communicate directly over one common medium. The medium can be either wired (*wired domain*) or wireless (*wireless domain*). Spatially overlapping wireless domains can for example be separated by choosing different center frequencies. There are different kinds of coupling stations between domains of different types. A *brouter* (from bridge/router) is an active station in two token-passing rings. In contrast, a repeater merely translates framing rules and may be used for example to couple a domain consisting only of wireless passive stations into a wired segment with active stations.

Multiple logical rings / segments can be created in different ways. In the *domain-driven MLR* approach there is a separate logical ring for each domain. Domains are coupled through brouters. If a wireless active station x is not reachable or loses the token, then only its own segment / logical ring is affected, the other segments remain functional. However, it may still not be acceptable to distort the operation of other wireless active stations in x's segment. Therefore, in the *wireless-master-driven MLR* approach each wireless active station runs its own token-passing ring / segment and is coupled to wired segments through its own brouter. A wireless domain with only passive stations is coupled through a repeater station. Finally, the *domain-group-driven MLR* approach can be used to group several wired and wireless domains with lots of inter-domain traffic into one segment / logical token-passing ring. Within a segment, only repeaters are used, different segments are coupled through brouters.

To support inter-segment transactions, proper addressing and routing mechanisms are needed. If a brouter behaves as a bridge, the routing is done on the basis of MAC addresses, therefore all the different segments form one single address space and thus there is a hard limit of 127 stations in the overall network, because the PROFIBUS does not support more addresses. The brouters have to maintain for each MAC address the “next hop interface”, i.e. the unique interface to which an incoming packet destined to a given MAC address is forwarded. In addition, the brouters have to check each packet on all interfaces whether these need to be forwarded. If a brouter behaves as a router, all stations need additional network layer addresses, but the MAC addresses may be re-used in every logical ring, allowing to form larger networks than possible with PROFIBUS.

The MLR approach has several advantages: token losses and transmission errors disturb only the operation in a single segment while leaving other segments fully operational; if several segments have mostly intra-segment traffic, they can transmit in parallel most of the time, increasing the capacity of the overall network; intra-segment communications does not need to cross media boundaries; and finally, it is possible to set certain network parameters (slot time, max_retry) different for each segment. However, the handling of inter-segment transactions is not without problems. First, if a packet needs to be relayed through brouters, there are not only the forwarding and propagation/transmission delays, but the brouters need also to acquire the token in the next-hop network before they can proceed. During this token waiting time further packets for the same next-hop network may arrive and must be queued, giving rise to additional queueing delays. Therefore, if the inter-segment traffic is not tightly controlled, it is hard to give any timing guarantees. It is also hardly possible to complete an inter-segment transaction without setting the initiators slot-time to very high values. Depending on the settings of the slot time and the max_retry parameter, it might be even impossible to complete the transaction before the initiating active station has exhausted all trials for its packet, which is interpreted as a failure. To prevent this, it is suggested to let the brouters send some "response-comes-later"-packets back to the initiating master and to send the final result when available. However, this requires additions to the protocol.

D. Virtual Ring Extension Solution

The virtual ring extension (VRE) approach discussed in [137] takes the results presented in Sections IV-B and IV-C regarding the real-time performance of the PROFIBUS token-passing protocol over wireless / error-prone links as a motivation to drop the PROFIBUS protocol on the wireless side. Instead, specifically tailored MAC- and link-layer protocols are used (for example polling-based protocols [139]), but the link layer interface to upper layers is the same as for the PROFIBUS protocol. On the wired side the unmodified PROFIBUS protocol is used. By this design, both wired and wireless stations offer the same set of link layer services to the upper layers. As a consequence, existing application layer software can run on both types of stations.

The coupling of wired and wireless segments is done by means of an intelligent station, the *base-station / interworking-unit* (BS-IWU). The BS-IWU acts *on behalf* of the wireless stations on the wired segment such that the wired stations perceive the wireless stations as normal wired stations. To achieve this, the BS-IWU has to create and accept token frames on the wired segment and to participate in the ring maintenance mechanisms as well, all on behalf of the wireless stations. Furthermore, the BS-IWU has to forward data frames between wired and wireless segments, and it serves as a wireless base station / central station for the wireless MAC protocol. It may optionally perform energy-proxy operations for certain types of wireless terminals, for example small, energy-constrained wireless sensors observing a slowly varying physical process, see Section III-E. This approach also allows to run a preemptive link layer protocol on the wireless link, the BS-IWU then translates between the protocols. Depending on the protocols the BS-IWU needs to keep some state to do this. In general, since protocol compatibility to the

PROFIBUS protocol is relaxed, it can be much easier to add mobility support and security features into the VRE approach.

While this approach allows to replace the token-passing protocol in the wireless segments by more appropriate protocols, there are also significant drawbacks. The BS-IWU is a fairly complex device and a single point of failure. Proper redundancy concepts have to be developed for this.

E. Application layer gateway solutions

Examples for the application layer gateway approach are described in [80] and [81]. In [80] a PROFIBUS network is augmented with wireless stations by means of a *protocol converter*, which acts as a PROFIBUS master on the wired part and as a *virtual master* on the wireless part. On the wireless side, an application layer instance runs a polling protocol utilizing IP datagrams on top of IEEE 802.11 with distributed coordination function (DCF) (*virtual polling algorithm*), effectively eliminating contention/collisions among wireless stations. The protocol converter puts all the packets destined to a certain (wired or wireless) station into a separate message queue and transmits them on the next occasion (defined by the length of the polling cycle and the allowable service time per station). In [81] a similar approach is described, with application to an automated container terminal. An experimental performance analysis evaluates throughput and round-trip times for different cross-traffic situations, polling cycle times and message queue lengths. It is shown that the ability to transmit packets within certain deadlines critically depends on the maximum allowable message queue lengths, with short lengths giving better deadlines at the expense of a certain amount of packet losses.

VII. WIRELESS FIP/WORLDFIP

A group at EPFL Lausanne has worked on transparent integration of wireless stations into the factory instrumentation protocol (FIP)/WorldFIP fieldbus [97], [98]. FIP/WorldFIP is a European fieldbus standard, which emerged from a french standard [131]. It uses a polling table to implement a real-time database: a special station, the *bus arbiter* broadcasts the identifier of a variable, and in turn the *producer* of this variable broadcasts its value. All interested stations (*consumers*) copy the variables value into an internal cache. The protocol has no support for mobile stations or for security. A more detailed description can be found in [29].

It is hard to construct a hybrid FIP/WorldFIP system, i.e. a system with both wired and wireless stations running in the same network. The protocol requires the producer to answer the bus arbiters requests within very short time (70 bit times), which is difficult to achieve if the (wired) bus arbiter polls a wireless producer connected by a repeater / bridge with some forwarding delay [98]. Therefore, the approach presented in [97] uses a wireless-to-wired gateway, which serves as central base station for the wireless part. The MAC protocol is based on a TDMA scheme. The base station is responsible for caching all process variables produced by wireless stations and to transmit these on the wired part, if requested. Conversely, the gateway also caches all process variables produced by wired stations and consumed by wireless stations, and broadcasts these on the wireless link. The case of asynchronous directed message transmission is not discussed, nor are transmission errors considered.

The european community project OLCHFA (June 92 until September 94) was one of the earliest attempts towards wireless fieldbus systems. The project was targeted at enhancing FIP/WorldFIP with wireless stations at the 2.4 GHz ISM band using a direct sequence spread spectrum (DSSS) physical layer [62]. However, the available publications put emphasis on the management of configuration data and on distributed algorithms for clock synchronization. Within the project a communications controller was developed, which can switch between using a wired and a wireless medium. The MAC and data link protocol of FIP/WorldFIP is not modified.

VIII. Controller Area Network (CAN)

There seems to be not so much visible work about wireless CAN systems. A group at the university of Sussex has worked on the topic of wireless CAN, with focus on MAC layer issues, not so much on integration issues. The CAN MAC protocol belongs to the class of priority arbitration protocols and is not directly implementable on wireless channels, as explained in Section IV-D. Two different approaches have been developed to resolve this problem. The WMAC approach uses CSMA with backoff times directly proportional to the message priority before starting to transmit [76], hence this approach has some similarities to the EY-NPMA protocol of HIPERLAN [37]. In the second approach the priority value is mapped onto the channel using an on-off-keying scheme: a station transmits a short burst, if the current priority bit is a logical one, or it switches to receive mode if it is a zero. If the station receives something in the receive mode, it gives up. The priority bits are handled from the most significant bit to the least significant bit, all stations have to be synchronized on bit boundaries [74]. In addition, this approach requires fast switching between transmit and receive mode for the radio modem.

IX. Other Systems

In this section we briefly touch efforts to bring wireless technology to further fieldbus systems. In [64] an application layer gateway is proposed to couple wireless stations into small so-called “islands of automation”, running the miniMAP protocol suite which covers the physical, data link and application layer (the full Manufacturing Automation Protocol (MAP) covers all seven OSI layers, and MAP 3.0 was the only commercially available protocol stack doing so). On the wireless side also a three-layer protocol is used. The transmission is based on CDMA. On the MAC layer two priorities are supported, one for time-critical traffic (type-I), the other one for time-insensitive data types (type-II). These are incorporated in a time-division frame structure such that a frame starts with a control period, followed by a time-period for type-I data and another time-period for type-II data. Time-sensitive data is subject to a handshake procedure similar to the RTS/CTS protocol employed in IEEE 802.11, testing whether the number of simultaneous CDMA transmitters is low enough to keep interference below a certain level. Type-II packets contend for the channel using an ALOHA-like protocol. The performance evaluation focuses on the properties of the wireless MAC protocol. For type-I packets the interference-induced loss probabilities and the average delay are computed under a Poisson traffic model, other channel errors are not taken into account.

For the IEC FieldBus [59] (which uses a centralized, polling-based access protocol for supporting periodic data and a token-passing protocol for asynchronous data) in [23] an architecture was proposed, which allows to couple several wired fieldbus segments using a wireless backbone based on IEEE 802.11 with point coordination function (PCF).

In [99] it is investigated, how the MAP/MMS application layer protocol [66] 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 [96] the same question was investigated with DECT as underlying technology. Again, time critical transmissions were not considered.

X. Further Issues for Wireless Industrial LANs

In this section we discuss some further interesting issues in wireless fieldbus systems.

A. TCP over Wireless Fieldbus

A fieldbus system does not only transmit real-time data, but also many time-insensitive data types like configuration data, program or result files, possibly from a server in a distant office network, running the TCP/IP protocol stack. A wireless fieldbus node running TCP is faced to two different problems:

- It is well-known that TCP has performance problems over error-prone wireless links, due to TCP's inability to differentiate between losses on the wireless link and losses through congestion in the network. TCP interprets losses as caused by congestion and reduces its transmission rate [11], [58], [25].
- Many fieldbus systems have small maximum packet sizes, for example, the PROFIBUS protocol allows a maximum of 246 user data bytes per packet. One major reason for this is to avoid situations where the transmission medium is blocked by a long packet while an important message is waiting (preemptibility). However, TCP segments have a header of at least 20 bytes length, and the header of the encapsulating IPv4 packet needs another minimum 20 bytes [123]. Different header compression schemes can be used to reduce header sizes, for example the SLIP header compression or the IETF robust header compression, which is especially designed for wireless links (RFC 3095 and subsequent ones). These schemes require setup of a "link layer connection" spanning the wireless hop, to negotiate the values of constant header fields and to introduce appropriate shorthands for these.

An approach solving these two problems at once is the Remote Socket Architecture (ReSoA) approach [118]. This approach uses a dedicated server station, the ReSoA server, which typically has a wired connection to the Internet, and also a (wireless or wired) connection to the other stations in its LAN ("local stations"). The motivation for this approach comes from the observation that virtually all applications using TCP/IP do so by means of the BSD socket interface instead of directly calling the TCP implementation. ReSoA splits the socket interface into two parts: the lower part of the socket interface and the TCP/IP protocol stack run on the ReSoA server, and the socket interface calls generated by the local stations are transported by a dedicated lightweight protocol (with low overhead) to the ReSoA server. Therefore, this approach has some similarities to remote procedure calls. This approach avoids spanning the TCP connection over the wireless link, and therefore TCP is not influenced by wireless errors.

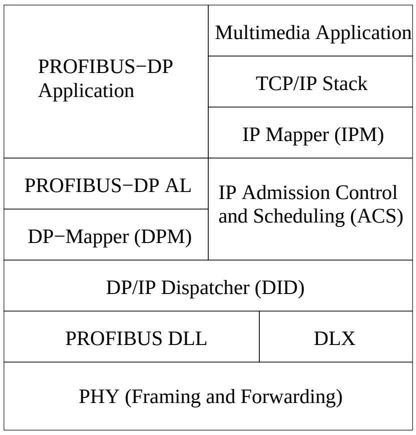

Fig. 8. RFieldbus Protocol Stack with IP/Multimedia Support (adapted from [113])

Clearly, there are other solutions [58], [11], ranging from pure link-layer approaches or TCP-aware link layer approaches (Snoop) to split-connection protocols and end-to-end solutions. All approaches, except the split-connection approaches [146], require transport of full TCP packets over the wireless link, including all the headers.

B. Multimedia

It is unquestionable that multimedia support gets more and more important for industrial networking [145], for example for video-based supervision of industrial processes [52], [113]. Therefore, fieldbus systems have to carry three different kinds of traffic: the (hard) real-time control traffic, the soft real-time multimedia traffic and best effort traffic like program files, emails, and so on. These have to be multiplexed onto the wireless link such that the timing requirements of the control traffic are not violated while the timing requirements of the multimedia traffic are satisfied to a sufficient degree. For the class of token-passing protocols (like PROFIBUS, IEEE 802.4 token-bus) the general suitability to carry multimedia traffic has already been investigated in [1], [102]. For the case of PROFIBUS (more specifically: PROFIBUS-DP), within the R-Fieldbus project an architecture was developed which allows to carry multimedia traffic in parallel to the PROFIBUS-DP control traffic. The relevant part of the protocol stack is shown in Figure 8, adapted from [113]. PROFIBUS-DP applications and the PROFIBUS-DP application layer remain unchanged, as well as the PROFIBUS data link-layer (DLL) and MAC protocol (part of DLL). The multimedia applications are supposed to run on top of the TCP/UDP/IP protocol stack. The IP mapper maps IP frames to PROFIBUS frames, including segmentation and reassembly. The IP admission control and scheduling (ACS) module classifies IP traffic from different applications into different priorities and restricts the bandwidth assigned to the respective classes. The PROFIBUS-DP Mapper (DPM) offers the same services and behaviour as the PROFIBUS DLL, but does not sit directly on the DLL. Instead, it assigns priorities to PROFIBUS-DP packets in the same way the ACS module does for IP packets. The DP/IP dispatcher (DID) is responsible for sharing the bandwidth between PROFIBUS-DP traffic and (important) IP traffic according to the classification done by the

ACS and DPM. An important goal for this is to share bandwidth such that the time-critical PROFIBUS-DP traffic is not distorted. The DID then provides the PROFIBUS-DLL with packets according to their priorities. The data link layer extensions (DLX) extend the PROFIBUS-DLL with radio-specific features.

XI. CONCLUSIONS AND OUTLOOK

In this paper we have surveyed the issues involved with creating wireless fieldbus systems. There are several challenges which have to be resolved, the most important ones being the issue of (hard) real-time communications over error-prone wireless links and the creation of hybrid systems, i.e. systems where both wired and wireless stations run in the same network. In addition, the introduction of wireless media does not only bring mobility as a benefit, but also the challenge to extend existing and non-mobility-aware systems with appropriate handover procedures. Security has also become an issue where it has not been one before: the open-ness of wireless media allows not only to eavesdrop or to inject malicious packets, distorting time-critical transmissions through jamming is also an option for an attacker.

So, are wireless media ready for use in safety- or even life-critical systems? If such a system needs a 100% (or very very close to 100%) guarantee that certain important messages are transmitted successfully within a bounded time, then at this point of time the available mechanisms to combat wireless channel errors seem not sufficient to achieve similar levels of reliability as wired systems (not to speak of fibre-cables with their extremely low error rates). However, often life-critical systems can fail in two ways [127]: unsafe or safe. Failing the unsafe way leads to a catastrophe, failing the safe way leads to a system shutdown into some state where the system is not operational but nobody is harmed. If the system designer takes explicitly into account that wireless transmissions can fail and if occasional safe system failures and subsequent restarts are acceptable, then wireless transmission may be an attractive option. The design goal for protocols and mechanisms is thus to make the times between safe failures longer and longer. We believe that acceptable service levels can be reached, if the existing approaches are refined and combined in suitable ways.

REFERENCES

[1] G. Agrawal, B. Chen, W. Zhao, and S. Davari, "Guaranteeing synchronous message deadlines with the timed token medium access control protocol," *IEEE Trans. on Computers*, vol. 43, no. 2, pp. 327 – 339, Mar. 1994.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: a survey," *Computer Networks*, vol. 38, pp. 393–422, 2002.

[3] I. F. Akyildiz, J. McNair, L. Carrasco, and R. Puigjaner, "Medium access control protocols for multimedia traffic in wireless networks," *IEEE Network Magazine*, vol. 13, no. 4, pp. 39–47, 1999.

[4] S. M. Alamouti, "A Simple Transmit Diversity Technique for Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 8, pp. 1451–1458, Oct. 1998.

[5] M. Alves, E. Tovar, and F. Vasques, "Evaluating the Duration of Message Transactions in Broadcast Wired/Wireless Fieldbus Networks," in *Proc. 27th Annual Conference of the IEEE Industrial Electronics Society (IECON'01)*. IEEE, Nov. 2001, pp. 243–248.

[6] M. Alves, E. Tovar, F. Vasques, G. Hammer, and K. Röther, "Mobility Management in Hybrid Wired/Wireless Token-Passing Fieldbus Networks," in *Proc. 2nd IFIP Networking Conference (Networking 2002)*, Pisa, Italy, May 2002.

[7] M. F. Alves, "Real-time communications over hybrid wired/wireless PROFIBUS-based networks," PhD Dissertation, University of Porto, Faculty of Engineering, Feb. 2003.

[8] M. H. Ammar and G. N. Rouskas, "On the performance of protocols for collecting responses over a multiple-access channel," *IEEE Trans. on Communications*, vol. 43, no. 2, pp. 412–420, Feb. 1995.

[9] A. Annamalai and V. K. Bhargava, "Analysis and Optimization of Adaptive Multicopy Transmission ARQ Protocols for Time-Varying Channels," *IEEE Trans. on Communications*, vol. 46, no. 10, pp. 1356–1368, 1998.

[10] C. M. Aras, J. F. Kurose, D. S. Reeves, and H. Schulzrinne, "Real-time communication in packet-switched networks," *Proceedings of the IEEE*, vol. 82, no. 1, pp. 122–139, Jan. 1994.

[11] H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. H. Katz, "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links," *IEEE/ACM Trans. on Networking*, vol. 5, no. 6, p. 756ff, 1997.

[12] H. Bengtsson, E. Uhlemann, and P.-A. Wiberg, "Protocol for wireless real-time systems," in *Proc. 11th Euromicro Conference on Real-Time Systems*, York, England, 1999.

[13] C. Bettstetter, G. Resta, and P. Santi, "The node distribution of the random waypoint mobility model for wireless ad hoc networks," *IEEE Trans. on Mobile Computing*, vol. 2, no. 3, pp. 257–269, July 2003.

[14] P. Bhagwat, P. Bhattacharya, A. Krishna, and S. K. Tripathi, "Using channel state dependent packet scheduling to improve TCP throughput over wireless LANs," *Wireless Networks*, vol. 3, no. 1, pp. 91–102, Mar. 1997.

[15] V. K. Bhargava and I. J. Fair, "Forward error correction coding," in *The Communications Handbook*, J. D. Gibson, Ed. Boca Raton, Florida: CRC Press / IEEE Press, 1996, pp. 166–180.

[16] U. Bilstrup and P.-A. Wiberg, "Bluetooth in Industrial Environment," in *Proc. 2000 IEEE Workshop on Factory Communication Systems, WFCS'2000*, Porto, Portugal, 2000, pp. 239–246.

[17] Bluetooth Consortium, "Specification of the bluetooth system," http://www.bluetooth.org, 1999.

[18] A. Boukerche, "Security and fraud detection in mobile and wireless networks," in *Handbook of Wireless Networks and Mobile Computing*, I. Stojmenovic, Ed. New York: John Wiley & Sons, 2002, pp. 309–323.

[19] E. H. Callaway, *Wireless Sensor Networks – Architectures and Protocols*. Boca Raton, Florida: Auerbach, 2003.

[20] R. Cam and C. Leung, "Multiplexed ARQ for time-varying channels – part I: System model and throughput analysis," *IEEE Trans. on Communications*, vol. 46, no. 1, pp. 41–51, Jan. 1998.

[21] ——, "Multiplexed ARQ for time-varying channels – part II: Postponed retransmission modification and numerical results," *IEEE Trans. on Communications*, vol. 46, no. 3, pp. 314–326, Mar. 1998.

[22] J. I. Capetanakis, "Tree Algorithm for Packet Broadcast Channels," *IEEE Trans. on Information Theory*, vol. 25, no. 5, pp. 505–515, Sept. 1979.

[23] S. Cavalieri and D. Panno, "On the integration of fieldbus traffic within IEEE 802.11 wireless LAN," in *Proc. 1997 IEEE International Workshop on Factory Communication Systems (WFCS'97), Barcelona (Spain)*, 1997.

[24] J. K. Cavers, *Mobile Channel Characteristics*. Boston, Dordrecht: Kluwer Academic Publishers, 2000.

[25] S. Cen, P. C. Cosman, and G. M. Voelker, "End-to-End Differentiation of Congestion and Wireless Losses," *IEEE/ACM Trans. on Networking*, vol. 11, no. 5, pp. 703–717, Oct. 2003.

[26] D. Chalmers and M. Sloman, "A survey of quality of service in mobile computing environments," *IEEE Communications Surveys and Tutorials*, vol. 2, no. 2, 1999, http://www.comsoc.org/livepubs/surveys.

[27] C.-F. Chiasserini and R. R. Rao, "Coexistence mechanisms for interference mitigation in the 2.4-ghz ism band," *IEEE Trans. on Wireless Communications*, vol. 2, no. 5, pp. 964–975, Sept. 2003.

[28] J.-D. Decotignie, "Wireless fieldbusses – a survey of issues and solutions," in *Proc. 15th IFAC World Congress on Automatic Control (IFAC 2002)*, Barcelona, Spain, 2002.

[29] J.-D. Decotignie and P. Pleineveaux, "A survey on industrial communication networks," *Ann. Telecomm.*, vol. 48, no. 9, p. 435ff, 1993.

[30] D. Duchamp and N. Reynolds, "Measured performance of wireless LAN," in *Proc. of 17th Conf. on Local Computer Networks, Minneapolis*, 1992.

[31] D. Dzung, C. Apneseth, G. Scheible, and W. Zimmermann, "Wireless Sensor Communication and Powering System for Real-Time Industrial Applications," in *Proc. 2002 IEEE Workshop on Factory Communication Systems, WFCS' 2002*, Västerås, Sweden, 2002, work-in-progress session.

[32] D. A. Eckhardt and P. Steenkiste, "A trace-based evaluation of adaptive error correction for a wireless local area network," *MONET - Mobile Networks and Applications*, vol. 4, pp. 273–287, 1999.

[33] M. Elaoud and P. Ramanathan, "Adaptive use of error-correcting codes for real-time communication in wireless networks," in *Proc. INFOCOM 1998*. San Francisco: IEEE, Mar. 1998.

[34] E. O. Elliot, "Estimates of error rates for codes on burst-noise channels," *Bell Systems Technical Journal*, vol. 42, pp. 1977–1997, Sept. 1963.

[35] A. Ephremides, "Energy concerns in wireless networks," *IEEE Wireless Communications*, vol. 9, no. 4, pp. 48–59, Aug. 2002.

[36] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Instrumenting the world with wireless sensor networks," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP 2001)*, Salt Lake City, Utah, May 2001.

[37] ETSI, *High Performance Radio Local Area Network (HIPERLAN) - Draft Standard*, ETSI, Mar. 1996.

[38] R. Fantacci and M. Scardi, "Performance evaluation of preemptive polling schemes and ARQ techniques for indoor wireless networks," *IEEE Trans. on Vehicular Technology*, vol. 45, no. 2, pp. 248–257, May 1996.

[39] L. Ferreira, M. Alves, and E. Tovar, "Hybrid Wired/Wireless PROFIBUS Networks Supported by Bridges/Routers," in *Proc. 2002 IEEE Workshop on Factory Communication Systems, WFCS'2002*, Västerås, Sweden, 2002, pp. 193–202.

[40] J. Flinn, "Extending mobile computer battery life through energy-aware adaptation," PhD Dissertation, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, Dec. 2001.

[41] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, vol. 40, pp. 311–335, Aug. 1998.

[42] Funbus-Projektkonsortium, "Das Verbundprojekt Drahtlose Feldbusse im Produktionsumfeld (Funbus) – Abschlußbericht," INTERBUS Club Deutschland e.V., Postf. 1108, 32817 Blomberg, Bestell-Nr: TNR 5121324, Oct. 2000, http://www.softing.de/d/NEWS/Funbusbericht.pdf.

[43] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Systems Technical Journal*, vol. 39, pp. 1253–1265, Sept. 1960.

[44] S. Glisic and B. Vucetic, *Spread Spectrum CDMA Systems for Wireless Communications*. Boston: Artech House, 1997.

[45] A. J. Goldsmith and S. B. Wicker, "Design challenges for energy-constrained ad hoc wireless networks," *IEEE Wireless Communications*, vol. 9, no. 4, pp. 8–27, Aug. 2002.

[46] A. C. V. Gummalla and J. O. Limb, "Wireless medium access control protocols," *IEEE Communications Surveys and Tutorials*, vol. 3, no. 2, 2000, http://www.comsoc.org/pubs/surveys.

[47] J. C. Haartsen, "The Bluetooth Radio System," *IEEE Personal Communications*, vol. 7, no. 1, pp. 28–36, Feb. 2000.

[48] D. Haccoun and S. Pierre, "Automatic repeat request," in *The Communications Handbook*, J. D. Gibson, Ed. Boca Raton, Florida: CRC Press / IEEE Press, 1996, pp. 181–198.

[49] D. Hampicke, A. Richter, A. Schneider, G. Sommerkorn, R. Thomä, and U. Trautwein, "Characterization of the Directional Mobile Radio Channel in Industrial Scenarios, based on Wide-Band Propagation Measurements," in *Proc. IEEE Vehicular Technology Conference*. Amsterdam, Netherlands: IEEE, Sept. 1999, pp. 3358–3362.

[50] B. A. Harvey and S. B. Wicker, "Packet combining systems based on the Viterbi decoder," *IEEE Trans. on Communications*, vol. 42, no. 2, pp. 1544–1557, Feb. 1994.

[51] O. Hersent, D. Gurle, and J.-P. Petit, *IP Telephony – Packet-based multimedia communications systems*. Harlow / England, London: Addison-Wesley, 2000.

[52] J. Hähniche and L. Rauchhaupt, "Radio communication in automation systems: the R-Fieldbus approach," in *Proc. 2000 IEEE International Workshop on Factory Communication Systems (WFCS 2000)*, Porto, Portugal, 2000, pp. 319–326.

[53] J. Hirai, T.-W. Kim, and A. Kawamura, "Practical study on wireless transmission of power and information for autonomous decentralized manufacturing system," *IEEE Trans. on Industrial Electronics*, vol. 46, pp. 349–359, Apr. 1999.

[54] 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)*. New Orleans, Louisiana: IEEE, 1994, pp. 1306–1311.

[55] T. Holliday, A. Goldsmith, and P. Glynn, "Wireless link adaptation policies: Qos for deadline constrained traffic with imperfect channel estimates," in *Proc. IEEE ICC 2002*. New York: IEEE, Apr. 2002.

[56] A. Honarbacht and A. Kummert, "WSDP: Efficient, yet reliable, transmission of real-time sensor data over wireless networks," in *Proc. Wireless Sensor Networks, First European Workshop (EWSN 2004)*, Berlin, Germany, Jan. 2004.

[57] I. Howitt, "Bluetooth performance in the presence of 802.11b WLAN," *IEEE Trans. on Vehicular Technology*, vol. 51, no. 6, pp. 1640–1651, Nov. 2002.

[58] H.-Y. Hsieh and R. Sivakumar, "Transport over wireless networks," in *Handbook of Wireless Networks and Mobile Computing*, I. Stojmenovic, Ed. New York: John Wiley & Sons, 2002, pp. 289–308.

[59] *IEC-1158-1, FieldBus Specification, Part 1, FieldBus Standard for Use in Industrial Control: Functional Requirements*, IEC - International Electrotechnical Commission.

[60] IEEE/ISO, *Information processing systems - Local Area Networks - part 4: Token-passing bus access method and physical layer specifications*, International Organization for Standardization, Aug. 1990.

[61] International Organization for Standardization, *ISO Standard 11898 - Road Vehicle - Interchange of Digital Information - Controller Area Network (CAN) for High-Speed Communication*, ISO - International Organization for Standardization, 1993.

[62] I. Izikowitz and M. Solvie, "Industrial needs for time-critical wireless communication & wireless data transmission and application layer support for time critical communication," in *Proc. Euro-Arch'93*. München: Springer Verlag, Berlin, 1993.

[63] J. A. Janet, W. J. Wiseman, R. D. Michelli, A. L. Walker, and S. M. Scoggins, "Using control networks for distributed robotic systems," in *Proc. IEEE International Conference on Robotics & Automation*, Detroit, Michigan, May 1999, pp. 1138–1143.

[64] S. Jiang, "Wireless communications and a priority access protocol for multiple mobile terminals in factory automation," *IEEE Trans. on Robotics and Automation*, vol. 14, no. 1, pp. 137–143, Feb. 1998.

[65] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, T. Imielinski and H. Korth, Eds. Dordrecht, Netherlands: Kluwer Academic Publishers, 1996, pp. 153–181.

[66] V. C. Jones, *MAP / TOP Networking – Achieving Computer Integrated Manufacturing*. New York: McGraw-Hill, 1988.

[67] H. ju Moon, H. S. Park, S. C. Ahn, and W. H. Kwon, "Performance Degradation of the IEEE 802.4 Token Bus Network in a Noisy Environment," *Computer Communications*, vol. 21, pp. 547–557, 1998.

[68] S. Kallel, "Analysis of a type-II hybrid ARQ scheme with code combining," *IEEE Trans. on Communications*, vol. 38, no. 8, pp. 1133–1137, Aug. 1990.

[69] V. K. Kongezos and C. R. Allen, "Wireless communication between A.G.V.'s (autonomous guided vehicle) and the industrial network C.A.N. (controller area network)," in *Proc. IEEE International Conference on Robotics & Automation*, Washington, DC, May 2002, pp. 434–437.

[70] C. Koulamas, S. Koubias, and G. Papadopoulos, "Real-Time Performance of Cut-Through Forwarding in Hybrid Wired/Wireless Industrial Networks," in *Proc. ETFA 03*, Lissabon, Portugal, 2003, pp. 23 – 30.

[71] C. Koulamas, A. Lekkas, G. Papandopoulos, G. Kalivas, and S. Koubias, "Delay performance of radio physical layer technologies as candidates for wireless extensions to industrial networks," in *Proc. IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'01)*, Sophia Antipolis, France, Oct. 2001, pp. 133–142.

[72] A. Köpke, A. Willig, and H. Karl, "Chaotic Maps as Parsimonious Bit Error Models of Wireless Channels," in *Proc. IEEE INFOCOM 2003*, 2003.

[73] J. F. Kurose, M. Schwartz, and Y. Yemini, "Multiple-access protocols and time-constrained communication," *ACM Computing Surveys*, vol. 16, pp. 43–70, Mar. 1984.

[74] 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*, Antalaya, Turkey, Nov. 1996, pp. 565–573.

[75] A. Kutlu, H. Ekiz, I. Erturk, and E. T. Powner, "Comparison of WMAC and COMB wireless protocols for control area network," in *Proc. 4th Communication Networks Symp.*, Manchester, UK, July 1997, pp. 254–256.

[76] 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*, Beijing, China, June 1996, pp. 494–499.

[77] ——, "Wireless control area network," in *Proc. Networking Aspects of Radio Communication Systems*, London, UK, Mar. 1996, pp. 3/1–3/4.

[78] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behaviour," *IEEE Trans. on Information Theory*, vol. 49, 2003, submitted.

[79] J. Lansford, A. Stephens, and R. Nevo, "Wi-fi (802.11b) and Bluetooth: enabling coexistence," *IEEE Network Magazine*, vol. 15, no. 5, pp. 20–27, Sept. 2001.

[80] K. C. Lee and S. Lee, "Integrated Network of PROFIBUS-DP and IEEE 802.11 wireless LAN with Hard Real-time Requirement," in *Proc. IEEE 2001 International Symposium on Industrial Electronics*, Korea, 2001.

[81] S. Lee, K. C. Lee, M. H. Lee, and F. Harashima, "Integration of mobile vehicles for automated material handling using Profibus and IEEE 802.11 networks," *IEEE Trans. on Industrial Electronics*, vol. 49, no. 3, pp. 693–701, June 2002.

[82] A. Lessard and M. Gerla, "Wireless communication in the automated factory environment," *IEEE Network Magazine*, vol. 2, no. 3, pp. 64–69, 1988.

[83] P. Lettieri and M. Srivastava, "Adaptive frame length control for improving wireless link throughput, range and energy efficiency," in *Proc. INFOCOM 1998*. San Francisco, CA: IEEE, 1998, pp. 564–571.

[84] S. Lin and D. J. Costello, *Error Control Coding – Fundamentals and Applications*. Englewood Cliffs, New Jersey: Prentice-Hall, 1983.

[85] C. Liu and J. Layland, "Scheduling algorithms for multiprogramming in a hard real-time environment," *Journal of the ACM*, vol. 20, no. 1, pp. 46–61, 1973.

[86] H. Liu, H. Ma, M. E. Zarki, and S. Gupta, "Error control schemes for networks: An overview," *MONET – Mobile Networks and Applications*, vol. 2, no. 2, pp. 167–182, 1997.

[87] J. W. S. Liu, *Real-Time Systems*. Upper Saddle River, NJ: Prentice-Hall, 2000.

[88] X. Liu and A. Goldsmith, "Wireless medium access control in networked control systems," in *Proc. IEEE American Control Conference*, 2004.

[89] N. P. Mahalik, Ed., *Fieldbus Technology – Industrial Network Standards for Real-Time Distributed Control*. Berlin: Springer, 2003.

[90] M. Malek, "Responsive computer systems," in *Encyclopedia of Microcomputers*, A. Kent and J. G. Williams, Eds. New York: Marcel Dekker, 1997, vol. 13, pp. 107–127.

[91] N. Malpani, Y. Chen, N. Vaidya, and J. Welch, "Distributed token circulation on mobile ad hoc networks," *IEEE Trans. on Mobile Computing*, vol. 4, 2004, accepted.

[92] A. M. Michelson and A. H. Levesque, *Error-Control Techniques for Digital Communication*. New York: John Wiley and Sons, 1985.

[93] L. B. Milstein and M. K. Simon, "Spread spectrum communications," in *The Communications Handbook*, J. D. Gibson, Ed. Boca Raton, Florida: CRC Press / IEEE Press, 1996, pp. 199–212.

[94] R. Min, M. Bhardwaj, S.-H. Cho, N. Ickes, E. Shih, A. Sinha, A. Wang, and A. Chandrakasan, "Energy-centric enabling technologies for wireless sensor networks," *IEEE Wireless Communications*, vol. 9, no. 4, pp. 28–39, Aug. 2002.

[95] E. Modiano, "An adaptive algorithm for optimizing the packet size used in wireless ARQ protocols," *Wireless Networks*, vol. 5, pp. 279–286, 1999.

[96] P. Morel, "Mobility in MAP networks using the DECT wireless protocols," in *Proc. 1995 IEEE Workshop on Factory Communication Systems, WFCS'95*, Leysin, Switzerland, 1995.

[97] P. Morel and A. Croisier, "A wireless gateway for fieldbus," in *Proc. Sixth International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 95)*, 1995.

[98] P. Morel, A. Croisier, and J.-D. Decotignie, "Requirements for wireless extensions of a FIP fieldbus," in *Proc. IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 96)*, Nov. 1996, pp. 116–122.

[99] P. Morel and J.-D. Decotignie, "Integration of wireless mobile nodes in MAP/MMS," in *Proc. 13th IFAC Workshop on Distributed Computer Control Systems DCCS95*, 1995.

[100] A. D. Myers and S. Basagni, "Wireless media access control," in *Handbook of Wireless Networks and Mobile Computing*, I. Stojmenovic, Ed. New York: John Wiley & Sons, 2002, pp. 119–143.

[101] A. Neskovic, N. Neskovic, and G. Paunovic, "Modern approaches in modeling of mobile radio systems propagation environment," *IEEE Communications Surveys and Tutorials*, vol. 3, no. 3, 2000, http://www.comsoc.org/livepubs/surveys.

[102] J. K.-Y. Ng, "MPEG transmission schemes for a timed token medium access control network," *ACM Computer Communication Review*, vol. 29, no. 1, pp. 66 – 80, Jan. 1999.

[103] G. T. Nguyen, , R. H. Katz, B. Noble, , and M. Satyanarayanan, "A trace-based approach for modeling wireless channel behavior," in *Proceedings of the Winter Simulation Conference*, Coronado, CA, December 1996.

[104] T. E. 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.

[105] J. R. Pimentel, *Communication Networks for Manufacturing*. Prentice-Hall International, 1990.

[106] P. Pleineveaux and J.-D. Decotignie, "Time critical communication networks: Field buses," *IEEE Network Magazine*, vol. 2, no. 3, pp. 55–63, 1988.

[107] G. J. Pottie and W. J. Kaiser, "Embedding the internet: Wireless integrated network sensors," *Communications of the ACM*, vol. 43, no. 5, pp. 51–58, May 2000.

[108] P. Raja, G. Noubir, L. Ruiz, J. Hernandez, and J.-D. Decotignie, "Analysis of Polling-Protocols for Fieldbus Networks," *ACM Computer Communication Review*, vol. 23, no. 3, pp. 69–90, 1993.

[109] T. S. Rappaport, "Characterization of UHF multipath radio channels in factory buildings," *IEEE Trans. on Antennas and Propagation*, vol. 37, no. 8, pp. 1058–1069, Aug. 1989.

[110] ——, "Indoor radio communications for factories of the future," *IEEE Communications Magazine*, vol. 27, no. 5, pp. 15–24, May 1989.

[111] ——, *Wireless Communications – Principles and Practice*. Upper Saddle River, NJ, USA: Prentice Hall, 2002.

[112] T. S. Rappaport, S. Y. Seidel, and K. Takamizawa, "Statistical channel impulse response models for factory and open plan building radio communication system design," *IEEE Trans. on Communications*, vol. 39, no. 5, pp. 794–807, May 1991.

[113] L. Rauchhaupt, "System and device architecture of a radio-based fieldbus - the RFieldbus system," in *Proc. Fourth IEEE Workshop on Factory Communication Systems 2002 (WFCS 2002)*, Vasteras, Sweden, 2002.

[114] S. Roundy, D. Steingart, L. Frechette, P. Wright, and J. Rabaey, "Power sources for wireless sensor networks," in *Proc. Wireless Sensor Networks, First European Workshop (EWSN 2004)*, Berlin, Germany, Jan. 2004.

[115] P. E. Rybski, S. E. Stoeter, M. Gini, D. F. Hougen, and N. P. Papanikolopoulos, "Performance of a distributed robotic system using shared communications channels," *IEEE Trans. on Robotics and Automation*, vol. 18, no. 5, pp. 713–727, Oct. 2002.

[116] C. Santivanez and I. Stavrakakis, "Study of Various TDMA Schemes for Wireless Networks in the Presence of Deadlines and Overhead," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 7, pp. 1284–1304, July 1999.

[117] G. Schäfer, *Security in Fixed and Wireless Networks – An Introduction to Securing Data Communications*. Chichester, UK: John Wiley and Sons, 2003.

[118] M. Schläger, B. Rathke, S. Bodenstein, and A. Wolisz, "Advocating a Remote Socket Architecture for Internet Access using Wireless LANs," *MONET - Mobile Networks and Applications*, vol. 6, no. 1, pp. 23–42, Jan. 2001.

[119] C. Schurgers, V. Raghunathan, and M. B. Srivastava, "Modulation scaling for real-time energy aware packet scheduling," in *Proc. Global Communications Conference (GlobeCom'01)*, San Antonio, TX, Nov. 2001, pp. 3653–3657.

[120] ——, "Power management for energy-aware communication systems," *ACM Transactions on Embedded Computing Systems*, vol. 2, no. 3, pp. 431–447, Aug. 2003.

[121] K. G. Shin and P. Ramanathan, "Real-time computing: A new discipline of computer science and engineering," *Proceedings of the IEEE*, vol. 82, no. 1, pp. 6–24, Jan. 1994.

[122] W. Stark, H. Wang, A. Worthen, S. Lafortune, and D. Teneketzis, "Low-energy wireless communication network design," *IEEE Wireless Communications*, vol. 9, no. 4, pp. 60–72, Aug. 2002.

[123] W. R. Stevens, *TCP/IP Illustrated Volume 1 - The Protocols*. Boston, Massachusetts: Addison-Wesley, 1995.

[124] H. Takagi, "Analysis and applications of a multi-queue cyclic service system with feedback," *IEEE Trans. on Communications*, vol. 35, no. 2, pp. 248–250, Feb. 1987.

[125] T. Takine, H. Takagi, and T. Hasegawa, "Sojourn Times in vacation and polling systems with Bernoulli feedback," *Journal of Applied Probability*, vol. 28, pp. 422–432, June 1991.

[126] The Editors of IEEE 802.11, *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, Nov. 1997.

[127] L. Tomek, V. Mainkar, R. M. Geist, and K. S. Trivedi, "Reliability Modeling of Life-Critical, Real-Time Systems," *Proceedings of the IEEE*, vol. 82, no. 1, pp. 108–121, Jan. 1994.

[128] W. Turin and R. van Nobelen, "Hidden Markov Modeling of Flat Fading Channels," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 9, pp. 1809–1817, Dec. 1998.

[129] E. Uhlemann, P.-A. Wiberg, T. M. Aulin, and L. K. Rasmussen, "Deadline-dependent coding - a framework for wireless real-time communication," in *Proc. International Conference on Real-Time Computing Systems and Applications*, Cheju Island, South Korea, Dec. 2000, pp. 135–142.

[130] *General Purpose Field Communication System, EN 50170, Volume 2: PROFIBUS*, Union Technique de l'Electricité, 1996.

[131] *General Purpose Field Communication System, EN 50170, Volume 3: WorldFIP*, Union Technique de l'Electricité, 1996.

[132] B. Walke, P. Seidenberg, and M. P. Althoff, *UMTS – The Fundamentals*. Chichester, UK: John Wiley and Sons, 2003.

[133] H. Wang and N. Moayeri, "Finite State Markov Channel - A Useful Model for Radio Communication Channels," *IEEE Trans. on Vehicular Technology*, vol. 44, no. 1, pp. 163–171, Feb. 1995.

[134] X. Wang and M. T. Orchard, "On reducing the rate of retransmission in time-varying channels," *IEEE Trans. on Communications*, vol. 51, no. 6, pp. 900–910, June 2003.

[135] P.-A. Wiberg and U. Bilstrup, "Wireless Technology in Industry – Applications and User Scenarios," in *Proc. ETFA 01*, Antibes-Juan les Pins, France, 2001, pp. 123–133.

[136] A. Willig, "Analysis of the PROFIBUS Token Passing Protocol over Wireless Links," in *Proc. 2002 IEEE International Symposium on Industrial Electronics*, L'Aquila, Italy, July 2002.

[137] ——, "An Architecture for Wireless Extension of PROFIBUS," in *Proc. IECON 03*, Roanoke, Virginia, 2003.

[138] ——, "Antenna Redundancy for Increasing Transmission Reliability in Wireless Industrial LANs," in *Proc. 9th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2003*, Lissabon, Portugal, 2003, pp. 7 – 14.

[139] ——, "Polling-based MAC Protocols for Improving Realtime Performance in a Wireless PROFIBUS," *IEEE Trans. on Industrial Electronics*, vol. 50, no. 4, pp. 806 –817, Aug. 2003.

[140] ——, "Wireless LAN technology for the factory floor," in *Handbook on Industrial Information Technology*, R. Zurawski, Ed. CRC Press, 2003, to appear.

[141] ——, "Exploiting Redundancy Concepts to Increase Transmission Reliability in Wireless Industrial LANs," *IEEE Trans. on Industrial Electronics*, 2004, accepted for publication.

[142] A. Willig and A. Köpke, "The Adaptive-Intervals MAC protocol for a wireless PROFIBUS," in *Proc. 2002 IEEE International Symposium on Industrial Electronics*, L'Aquila, Italy, July 2002.

[143] A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, "Measurements of a wireless link in an industrial environment using an IEEE 802.11-compliant physical layer," *IEEE Trans. on Industrial Electronics*, vol. 49, no. 6, pp. 1265–1282, 2002.

[144] A. Willig and A. Wolisz, "Ring stability of the PROFIBUS token passing protocol over error prone links," *IEEE Trans. on Industrial Electronics*, vol. 48, no. 5, pp. 1025–1033, Oct. 2001.

[145] C.-H. Wu, J. D. Irwin, and F. F. Dai, "Enabling Multimedia Applications for Factory Automation," *IEEE Trans. on Industrial Electronics*, vol. 48, no. 5, pp. 913–919, Oct. 2001.

[146] R. Yavatkar and N. Bhagwat, "Improving end-to-end performance of tcp over mobile internetworks," in *Proc. Workshop on Mobile Computing Systems and Applications*, Dec. 1994.

[147] H. Ye, G. C. Walsh, and L. Bushnell, "Real-time mixed traffic wireless networks," *IEEE Trans. on Industrial Electronics*, vol. 48, no. 5, Oct. 2001.

[148] Z. Zhang and A. S. Acampora, "Performance of a modified polling strategy for broadband wireless lans in a harsh fading environment," *Telecommunication Systems*, vol. 1, pp. 279–294, 1993.

[149] M. Zorzi and R. R. Rao, "Error Control and Energy Consumption in Communications for Nomadic Computing," *IEEE Trans. on Computers*, vol. 46, pp. 279–289, Mar. 1997.

PLACE
PHOTO
HERE

Andreas Willig (M '97) is currently an assistant professor with the Hasso-Plattner-Institute for Software Systems Engineering at the University of Potsdam, Germany. He obtained the Dr.-Ing. degree in electrical engineering from Technical University Berlin (Germany) in 2002, and the diploma degree in computer science from University of Bremen (Germany) in 1994. His research interests include wireless networks, fieldbus and real-time systems, ad-hoc and sensor networks, with specific focus on performance aspects.

PLACE
PHOTO
HERE

Adam Wolisz (SM) is currently a Professor of Electrical Engineering and Computer Science 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 research interests are in architectures and protocols of communication networks as well as protocol engineering with impact on performance and Quality of Service aspects.