Achieving Hybrid Wired/Wireless Industrial Networks with WDetServ: Reliability-Based Scheduling for Delay Guarantees

Samuele Zoppi, Amaury Van Bemten, H. Murat Gürsu, Mikhail Vilgelm, Jochen Guck, Wolfgang Kellerer

***Abstract*—Industrial control systems are foreseen to operate over hybrid wired/wireless networks. While the controller will be deployed in the wired network, sensors and actuators will be deployed in a Wireless Sensor Network (WSN). To support QoS for control systems, an end-to-end *delay bound* and a *target reliability* must be provided in both wireless and wired domains. However, for industrial WSNs, guaranteeing reliability is a challenging task because of low-power communications and the harsh wireless environment. In this work, we present the first QoS framework for arbitrary hybrid wired/wireless networks, which guarantees that the delay bound and the target reliability of each application are provided. As part of this framework, we propose the first reliability-based scheduler for WSN able to achieve a target reliability in the presence of dynamic interference. Simulations of the proposed scheduler prove its suitability in different interference scenarios and motivate further work.**

I. INTRODUCTION

A typical next-generation industrial automation system consists of different levels [1], [2]: the field level, where sensors and actuators are directly involved in the automation process, the automation level, where process control decisions are taken by industrial controllers such as programmable logic controllers, and the management level, where best-effort IP traffic is exchanged. Automation and management levels are typically connected to wired networks, while devices in the field level are wirelessly interconnected by an Industrial Wireless Sensor Network (IWSN).

To support the communication requisites between controller and field devices, strict industrial Quality of Service (QoS) requirements must be satisfied. We define industrial QoS as the *delay bound* and *target reliability* of an application. These requirements must be satisfied in both wired and wireless network domains.

Despite the large number of solutions for providing industrial QoS in hybrid wired/wireless networks, there is still no *seamless* industrial QoS provisioning across wired and wireless domains. Existing solutions mainly consist in proprietary implementations [2], and the interconnection between wired and wireless networks is not given by design. To interconnect arbitrary wired and wireless networks, a hybrid QoS-aware interface is needed and missing in the literature. Focusing on wired industrial communications, Guck *et al.* [3]–[5] proposed a routing framework for seamless industrial QoS provisioning. The framework abstracts the network as a generic model, over which end-to-end QoS routing is performed. The concrete QoS aspects of this model are implemented via an interface. In their work, they only propose DetServ, an implementation of this interface for wired packet switched IP networks to provide delay guarantees. To extend their solution to a hybrid industrial network, a wireless DetServ (WDetServ) model is needed which fulfills the *delay bounds* and *target reliability* with wireless communications. In fact, both models can coexist in the framework, and end-to-end QoS routing can be performed jointly in wireless and wired networks.

Furthermore, unlike wired links which are assumed to be 100% reliable, wireless links are naturally subject to packet losses. To overcome this issue in TDMA wireless networks, dynamic scheduling of communication resources is needed, which reacts to changes in the channel quality. This is particularly challenging for IWSN due to the low-power nature of the communication and the harsh industrial wireless environment. Thanks to the IEEE Std. 802.15.4e (TSCH) radio resources can be abstracted in a time-frequency resource grid, and resource blocks (RBs) can be allocated to industrial applications selecting specific time and frequency values. Different RBs have different, time-varying Packet Delivery Ratios (PDR), which might not meet the reliability requirements of the application. State-of-the-art solution to this problem is blacklisting [6], i.e. excluding interfered channels from the RBs, thus reducing the capacity of the system. To overcome the capacity limits, a central scheduler is needed which is able to compute a schedule fulfilling the *target reliability* combining RBs and not excluding them. Although several scheduling techniques for TSCH have been proposed, there is no scheduler which supports reliability guarantees efficiently.

A. Contributions

In this article we solve the issue of seamless end-to-end industrial QoS provisioning in hybrid wired/wireless networks. We achieve this by means of two contributions:

- By extending the QoS framework presented in [4] with the WDetServ model, we provide the first framework for seamless QoS provisioning in arbitrary hybrid wired/wireless industrial networks.
- We propose the first reliability-based dynamic scheduler for TSCH IWSNs which is able to guarantee a *target reliability* in presence of dynamic interference. Reliability is provided combining re-transmissions over interfered channels efficiently using all available radio resources.

This work has received funding from the German Research Foundation (DFG) grant KE1863/5-1 as part of the priority program SPP 1914 Cyber-Physical Networking and from the European Unions Horizon 2020 research and innovation programme under grant agreement No. 671648 (VirtuWind).

(Corresponding author: Samuele Zoppi) The authors are with the Chair of Communication Networks, Technical University of Munich, 80333 Munich, Germany (e-mail: {samuele.zoppi, amaury.van-bemten, murat.guersu, mikhail.vilgelm, guck, wolfgang.kellerer}@tum.de).

B. Related Work

1) Hybrid Wired and Wireless Networks: Research in industrial communication networks has always pursued the integration of wired and wireless networks in order to provide end-to-end industrial QoS. For this reason, a considerable number of hybrid solutions merging wireless and wired in industrial networks has been developed.

The existing work mainly covers the implementation aspects of integrating the existing wired standards to the different wireless communications [7]–[17]. In this sense, as discussed in the survey [18], several types of integrations are possible, but require *modifications* to the underlying systems. A popular solution is the integration using a gateway. The gateway can, for instance, forward the messages at the IP layer, or interconnect the systems at the medium access or application layer. Furthermore, in [19], the detailed description of different interconnection types is presented, but the problem of QoS guarantees is highlighted as the major issue while integrating wireless with wired communications.

In the direction of seamless integration of arbitrary wired and wireless technologies, the IETF working group 6TiSCH has introduced the 6TiSCH architecture to enable deterministic IP-based industrial wireless communications [20]. Their architecture interconnects the wired and wireless domains and enables the ability of providing QoS in a Software-Defined way [21] by means of scheduling. In [21], it is shown that their solution can provide integration to WSN in an optimal centralized way. However, their work shows integration methods between deterministic routing and scheduling in WSN, but does not provide a complete solution for hybrid end-to-end routing with QoS guarantees.

In summary, none of the existing methods provides an end-to-end QoS framework for provisioning of industrial QoS on arbitrary wired/wireless networks.

2) QoS Provisioning in Wireless Networks: A traditional approach for QoS provisioning in wireless networks is the joint optimization of different reliability-enhancing techniques across different layers. For instance, in [22] different modulation and transmission techniques are analyzed for cluster-based WSN, while, in [23], link layer QoS in mobile networks is improved modeling the effects of MIMO diversity schemes and Adaptive Modulation and Coding.

In general, several techniques aimed at improving QoS have been developed on link, network and transport layers as surveyed by [24] and [25]. Recently, a promising approach for QoS improvements in wireless networks is to exploit diversity through redundancy. In [26] the spatial redundancy of several WSN Access Points is exploited to achieve lower packet error rates. A similar concept is deployed in Wi-Red [27], a seamless link-level redundancy technique for Wi-Fi networks based on time and frequency diversity. Besides and complementary to the redundancy techniques, QoS can be provisioned via appropriate management of wireless resources through scheduling.

In general, scheduling of wireless resources is an NP-hard combinatorial problem [28] of mapping the demands to the time-frequency resources. In contrast to IWSN, it is a mature research topic in cellular networks, such as 3GPP Long Term Evolution (LTE). The scheduling problem in LTE is typically considered as a trade-off between resource utilization and fairness. Schedulers in LTE could be classified into two types: best effort and quality of service aware. *Maximum throughput (MT)* and *proportional fair (PF)* are classic examples of the first type. MT allocates the resources always to the users with the best channel quality, hence, maximizing the total resource utilization under the demand saturation assumption. On the other hand, PF weights the current channel quality and the transmission history of a user, achieving a balance between fairness and utilization. QoS-aware schedulers are typically based on the prioritization of packets with strict delay or data rate requirements [29]–[31]. However, in LTE, reliability is outside of the scheduling problem's scope, as it is achieved by other techniques, such as hybrid automatic repeat request and adaptive modulation and coding.

Since the release of the TSCH Medium Access and the 6TiSCH architecture [20], considerable amount of work was performed concerning the problem of scheduling in IWSN. Different distributed schedulers were proposed in TSCH [32]–[40]. However, most of the de-centralized solutions provide dynamic traffic-based scheduling in a multi-hop network [41]. Here reliability is investigated only as a result of internal interference caused by frequency reuse. Furthermore, several centralized traffic-based schedulers have been studied for TSCH [42]–[49]. Also in this case, the scheduling algorithms allocate resources based on the amount of traffic, and aim at minimizing the latency and the buffer occupation. As centralized schedulers build conflict-free schedules, there is no packet loss due to internal interference, however the effect of external interference is not taken into account.

The state-of-the-art solution to external interference in order to guarantee reliability is blacklisting, i.e. excluding interfered channels. However, Zoppi *et al.* [50], [51] showed that reliability can be improved by using all channels independent of the interference scenario.

In summary, none of the existing solutions provide a reliability-based scheduler for IWSN able to provide a target reliability in presence of dynamic interference.

C. Structure of the Article

The remainder of the article is structured as follows. In Sec. II, the end-to-end QoS routing framework is presented together with the first contribution, WDetServ. In Sec. III, the reliability-based wireless dynamic scheduler is introduced. In Sec. IV, the details of the implementation of reliability-based wireless scheduler are discussed, and, in Sec. V, evaluated by means of simulations in an interfered IWSN. Finally, in Sec. VI, the conclusions are drawn and future work is discussed.

II. HYBRID WIRED-WIRELESS INDUSTRIAL QOS FRAMEWORK

In this section, we present the first contribution of this article, WDetServ, the wireless extension of the routing framework presented in [4]. In Sec. II-A, we define the Quality of Service (QoS) requirements of industrial applications. Then,

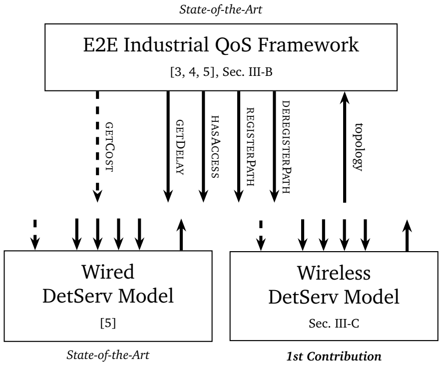

Fig. 1: Illustration of the first contribution of this article. We propose a DetServ model for wireless communication links (*WDetServ Model*) compliant with the interface defined by Guck *et al.* [3]–[5] so that industrial QoS in hybrid wired/wireless environments can be provided.

in Sec. II-B, we introduce a state-of-the-art framework for the end-to-end provisioning of such industrial QoS in wired networks [3], [4]. The framework requires a model of the underlying network. The interaction between the framework and underlying models is illustrated in Fig. 1. In Sec. II-C, we define WDetServ, a model for wireless communications that accommodates the requirements of the industrial framework and hence provides industrial QoS in wireless environments. In Sec. II-D, we discuss how both wired and wireless models can be jointly used in order to provide end-to-end industrial QoS in hybrid environments.

A. Industrial QoS: Definition

Industrial networks carry critical messages, e.g., measurement and actuation signals for large automated production facilities. Most of these critical messages must be delivered within a given time; otherwise, actuators can misbehave, thereby potentially leading to important material or physical damage [2], [52], [53]. As such, we define the industrial QoS requirements of an application by means of two parameters: its *delay bound* and its *target reliability*. The target reliability defines the percentage of sent packets that have to reach the destination within the delay bound.

B. State-of-the-Art: SDN-based Industrial QoS Framework

Focusing on wired industrial communications, Guck *et al.* [3]–[5] proposed an online routing framework for the rapid provisioning of industrial QoS. The framework leverages the centralized knowledge offered by Software-Defined Networking (SDN) to assume that an accurate model of the state of the network can be kept in the control plane. For each application request, the framework runs a routing procedure which relies on a *model* of the underlying network. The model is responsible for ensuring that, if the routing procedure appropriately uses its interface, the requirements of all applications will be met at any time. Once a route has been found, the model is updated in order to reflect the consumption of resources of the application.

As such, Guck *et al.* [5] identify the requirements of a network model. First, the network model must provide the routing procedure a topology on which to perform path finding. Second, the network model must provide the following interface to the routing procedure (see Fig. 1):

- GETDELAY. It provides, for each edge in the topology, the delay of this edge. It allows the routing procedure to ensure that it provides a solution respecting the delay bound of the application.
- HASACCESS. It checks whether or not there are still enough resources available at a given edge. It is used to ensure that the new application will not violate the delay bound of previously accepted applications in the network.
- REGISTERPATH. After the routing procedure is completed, it updates the network model state to reflect the addition of the new application.
- DEREGISTERPATH. It updates the network model state to reflect the removal of an application.

Additionally, a GETCOST function can be defined for each edge in order to provide to the routing procedure a way of knowing which solution to prefer if several are available. The GETCOST function should be defined so as to maximize the amount of applications that can be accepted. The routing procedure then tries to find the path with minimal cost that satisfies the delay constraint. The resulting routing problem corresponds to a *Constraint Shortest Path* (CSP) problem [54].

C. Contribution: Wireless DetServ Network Model

The industrial QoS framework is independent of the network model (see Fig. 1). As such, in order to provide a hybrid wired/wireless industrial QoS framework, it is sufficient to provide a network model for the wireless domain that satisfies the requirements described in Sec. II-B.

Authors in [5] proposed two network models for wired infrastructures. Both models rely on deterministic network calculus theory [55], [56] to ensure that the delay bounds of the different applications are never exceeded. For this, each application has to specify its maximum *burst* and its maximum *sustainable rate* [56].

In this section, we present the proposed *wireless DetServ* (WDetServ) model, a network model for TDMA-based wireless communications that can be used as part of the framework described in Sec. II-B. In the WDetServ model, the maximum packet size, the minimum packet size, the maximum burst and the sustainable rate of the application have to be known. Unlike wired links, which are assumed to be 100% reliable, wireless links are naturally subject to packet losses that cannot be neglected. For this reason the model relies on a reliability-based scheduler that will be described in Sec. III.

First, in Sec. II-C1, we describe the context in which the proposed network model can be used. Second, in Sec. II-C2, we describe the topology provided by the model to the routing procedure of the QoS framework. Third, in Sec. II-C3, we describe the implementation of the interface shown in Fig. 1.

1) Setup: Independent Wireless Gateways (WGs): We consider a setup in which WSN motes are connected to the wired infrastructure via *wireless gateways* (WGs) in a star topology fashion. This topology is particularly common in a dense industrial scenario where gateways are extensively installed

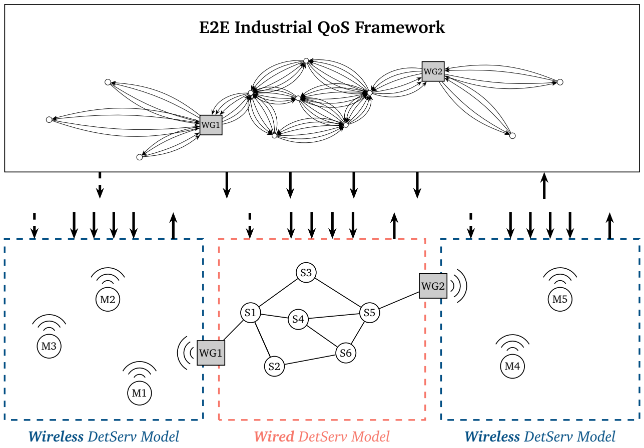

Fig. 2: Graphical representation of the integration of our proposed *WDetServ model* with the wired DetServ models proposed by Guck *et al.* [5].

across the factory (WISA [57], WirelessHart [58]) and it is suitable for low-latency applications as control messages can be delivered faster. In fact, multiple hops negatively affect latency and are more difficult to organize.

Each mote communicates solely with the WG to which it is connected and we assume that motes connected to different WGs do not interfere with each other. This can for instance be achieved during the network planning phase by appropriately placing the WGs, or with coordinated scheduling. Coordinated scheduling in a scenario where one mote can reach several WGs could bring several benefits such as, for instance, improved reliability thanks to redundant links, or support to mobile motes. However, due to space constraints, the discussion and evaluation of coordinated scheduling is left as future work.

As a result, a WDetServ model operates independently for each WG in the network. This is illustrated in Fig. 2. While the wired part of the network, including the WGs, is modeled by a unique wired DetServ model, the different wireless links associated to each WG are modeled by independent WDetServ models. Hence, in the following, we discuss the operation of a single WDetServ model.

2) Routing Topology: Frame Size Link Topology: Every TDMA wireless link is modeled as a set of directed edges defining the different frame sizes that can be used. In order to avoid ambiguity, we refer to these frames as *sub-frames*, in contrast to the main TDMA *super-frame*. The size of every sub-frame constraints the maximum delay and reliability that can be achieved using this sub-frame.

Indeed, since a smaller sub-frame size reduces the number of transmission opportunities, it ensures a smaller delay at a price of a reduced reliability. The TDMA super-frame of length s_0 is split into s_i $(i = 1, \dots n)$ different TDMA sub-frames of different sizes. In order for the different TDMA sub-frame sizes to coexist at the link layer, the n different sub-frame sizes have to be integral divisors of s_0. For example, if $s_0 = 8$, the possible sub-frames sizes are 1, 2, 4 and 8 slots. Route selection on this topology hence defines the sub-frame size to be used for the wireless medium access. We define this topology as the *frame size link topology* of the wireless network. This is illustrated in Fig. 2 for $s_0 = 2$. Since $s_0 = 2$, the possible sub-frame sizes are 1 and 2 slots, thereby defining two directed links for both the uplink and downlink between the motes and their respective WGs.

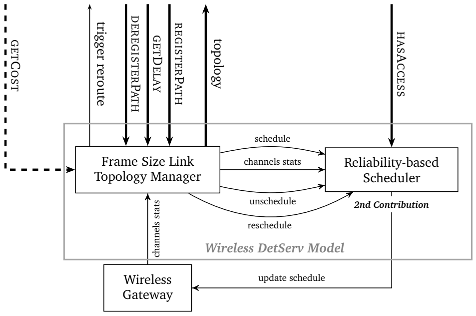

Fig. 3: Internal representation of the proposed *WDetServ model*. The model exposes the *frame size link topology* to the end-to-end QoS framework and implements the interface described by Guck *et al.* [3]–[5].

3) Implementation of the Model: In this section, we describe in details how the interface of the WDetServ model is implemented. This is illustrated in Fig. 3.

a) topology: The topology provided to the routing procedure of the industrial QoS framework is the frame size link topology described in Sec. II-C2.

b) GETDELAY*:* For each edge of the frame size link topology, the GETDELAY function returns the worst-case delay of a packet transmission using the sub-frame size corresponding to the given edge. The transmission of a packet lasts up to one sub-frame length $s_i t_s$, where t_s is the duration of a slot. However, considering that a burst of B packets could arrive just after the beginning of a new frame, each packet would have to wait for B additional frames to be fully transmitted. As such, the worst-case delay of a packet is

$$\mathbf{T} = (B + 1)(s_i t_s), \tag{1}$$

where B is the maximum burstiness of the application. As resources are assumed to be correctly allocated by the scheduler, this bound will be always satisfied for each application.

c) (DE)REGISTERPATH*:* A scheduler is responsible for providing reliability guarantees through appropriate scheduling. Upon receiving a REGISTERPATH (resp. DEREGISTERPATH) call, the scheduler schedules (resp. unschedule) the given application according to its reliability level to the sub-frame corresponding the given edge (*schedule* (resp. *unschedule*) arrows in Fig. 3). The result is then forwarded to the WG in order to notify the involved motes (*update schedule* arrows in Fig. 3). In order to evaluate the reliability of a given schedule, channel statistics must be gathered. This information is obtained by the WG and forwarded to the scheduler (*channel stats* arrows in Fig. 3). Upon any channel statistics change, applications reliability requirements can get violated. In this case, the scheduler tries to reschedule the applications such that their respective target reliability values are met again (*reschedule* arrow in Fig. 3). If the rescheduling succeeds, the WG receives and forwards the new schedules to the devices. If the rescheduling fails, end-to-end rerouting is necessary (*trigger reroute* arrow in Fig. 3). Indeed, changing the paths taken in the wired infrastructure will potentially allow to use other sub-frame sizes on the wireless link, thereby potentially making the application schedulable again. The concept of end-to-end routing through the wireless and wired links will be further discussed in Sec. II-D2. The operations of the scheduler correspond to the second main contribution of this article and are detailed in Sec. III.

d) HASACCESS*:* It verifies that communication resources are available for a given application checking that:

1) A schedule that provides the target reliability and delay bound is available. In a single-hop wireless communication this check is sufficient for providing end-to-end reliability. Indeed, since the wired links are assumed to provide 100% reliability (see Sec. II-C), the end-to-end reliability of a path corresponds to the reliability of its wireless hop.
2) Enough data rate is available. As an application sends at most one packet per frame, the lowest achievable data rate corresponds to its minimum packet size divided by the duration of the frame. That is, $r_a \leq l_a/(s_i t_s)$, where r_a and l_a respectively represent the data rate and the minimum packet size of an application a.
3) The packet size of the application is smaller than the maximum packet size B_s that can be sent in a single slot. That is $l_a^{max} \leq B_s$, where l_a^{max} is the maximum packet size of an application a.

e) GETCOST*:* The cost function is not defined by the model. It can depend on any parameters such as the channel statistics, the amount of scheduled motes, the delay of the edges and the size of the frames. The investigation and evaluation of the impact of the cost function is left for future work.

D. Combining the Wireless and Wired Models

1) Merging the Queue Link and Frame Size Link Topologies: In order for the routing procedure to operate on a single topology, the topologies provided by both the wired and wireless models have to be merged. This operation is illustrated in Fig. 2. Indeed, as each WG appears once in both DetServ [5] and WDetServ topologies, the final topology is simply obtained by merging the individual topologies at the different WGs.

2) The Need for a Combined Routing Decision: Benefits: In this section, we briefly highlight, with three practical use cases, the benefits of applying the routing procedure simultaneously on the combined wired and wireless links.

a) Optimal Backbone Usage: Typical per-hop delays in industrial environments are in the order of milliseconds [5]. Line or ring topologies, typical for industrial scenarios, can reach up to tens of hops. As a result, delays can reach up to tens of milliseconds in the wired backbone. In the wireless Std. IEEE 802.15.4e, a typical time slot duration is $t_s = 10$ ms [59]. From Eqn. (1), considering the smallest frame ($s_3 = 1$), this leads to delays of several tens of milliseconds. Consequently, the delay of the wireless link is comparable to the delay in the backbone. As such, routing decisions in the wired network influence the remaining choices in the wireless links for providing the required delay bounds, and vice-versa. Hence, applying the routing procedure globally allows to adapt the delay in the wired backbone such that the delay bound can be provided by the available delays at the wireless links.

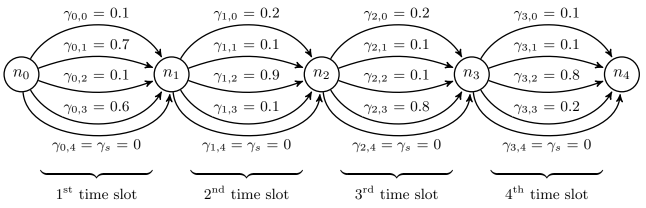

Fig. 4: Example of a *scheduling graph* used by our proposed scheduler in order to allocate resource blocks (RBs) to applications.

b) Reroutes upon Channel Quality Updates: The quality of the wireless channel changes, and this potentially leads to the violation of previously provided guarantees. In such a case, a global routing decision allows to reroute an application in the backbone (e.g., taking a faster path) such that the higher delay of the wireless link is still sufficient to fulfill the requirements of the application (*trigger reroute* arrow in Fig. 3).

c) Multiple WGs: A mote could have the opportunity of choosing between different WGs in its vicinity. Global routing allows to favor the usage of one WG over another in order to be able to fulfill the requirements of the application.

III. RELIABILITY-BASED SCHEDULER

In this section we present our second contribution, the reliability-based dynamic scheduler. In fact, as discussed in Sec. II-C, a wireless scheduler is needed that provides reliability over lossy wireless links, and enables seamless QoS provisioning in hybrid networks.

We model the wireless communication resources by means of a time-frequency grid, assuming a TDMA and FDMA combined medium access. The combination of a time slot and a frequency is called *resource block* (RB) and it is identified, here, by the time-frequency pair (t, f). By means of a *schedule*, it is possible to allocate different RBs from the time-frequency grid to an application. The *Packet Delivery Ratio* (PDR) of a RB corresponds to the probability of a successful transmission in this RB and is defined as $\gamma_{t,f}$.

First, we describe how we model the time-frequency grid as a graph (Sec. III-1). Second, we explain how the scheduling problem can be mapped into a *Constrained Shortest Path* (CSP) problem (Sec. III-2). Third, we present an algorithm able to solve the reliability-based scheduling problem (Sec. III-3). Finally, in Sec. III-4, we show how the proposed reliability-based scheduler can be integrated in the WDetServ model presented in Sec. II-C.

1) Graph Definition: The proposed scheduler is based on a graph model, the *scheduling graph* (see Fig. 4). As described in Sec. II-C3, one scheduler operates for each WG, i.e., one scheduling graph is defined per WG. Nodes in the scheduling graph represent time instants before and/or after the different time slots. Edges represent the transmission at different frequencies and in different time slots. For each time slot, one additional edge, the *silent edge*, represents no transmission. In the example of Fig. 4, for each time slot, the four upper edges represent transmission at four different frequencies and the lower edge corresponds to the silent edge. That is, every possible RB is represented by an edge in the scheduling graph. As a result, path finding on this graph from the first to the last node defines a schedule. The PDR of the RBs are obtained from the channel statistics gathered by the WG and represent the weights of the different edges (see Sec. II-C and Fig. 3). As the silent edges never lead to a successful transmission, their PDR is defined as $\gamma_s = 0$.

2) Formulation as a Constrained Shortest Path (CSP) Problem: The proposed scheduler operates by finding a path in the scheduling graph such that the reliability of the found path, i.e., the product of the unavailabilities $(1 - \gamma_{t,f})$ of the chosen edges, satisfies the target reliability requirement of the application r_0, i.e.,

$$\prod_{(t,f)\in S} 1 - \gamma_{t,f} < 1 - r_0, \tag{2}$$

where S is the set of edges constituting the found path, i.e. the schedule.

The unavailability metric is multiplicative, i.e., the metric values of the individual edges of a path have to be multiplied in order to obtain the end-to-end metric value of the path. However, most state-of-the-art routing algorithms can only deal with additive metrics [54], i.e., with metrics whose end-to-end value corresponds to the sum of their values on the individual edges. As a result, transforming the unavailability metric into an additive metric allows to use state-of-the-art algorithms. To do so, we apply the logarithm operator to Eqn. (2). This successively yields[1]

$$\log\left(\prod_{(t,f)\in S} 1 - \gamma_{t,f}\right) < \log(1 - r_0), \tag{3}$$

$$\sum_{(t,f)\in S} \log\left(1 - \gamma_{t,f}\right) < \log(1 - r_0), \tag{4}$$

There are potentially more than one path in the scheduling graph satisfying Eqn. (4), i.e., there are potentially more than one schedule that allocate enough resources for a communication in order to satisfy its target reliability value. In order to define a preference order among the schedules, we define a cost $c_{t,f}$ for each edge in the scheduling graph. The cost function $c_{t,f}$ can be defined arbitrarily and, in general, should be defined in a way that leads to the greatest amount of schedulable applications. An example of cost function defines $c_s = 0$ for silent edges and $c_{t,f} = 1$ for all other edges. This function leads to the assignment of the minimum amount of RBs to each application.

The routing problem for finding a schedule then consists in solving

$$\min_{S\in\mathcal{P}} \sum_{(t,f)\in S} c_{t,f} \tag{5}$$

$$\text{s.t.} \sum_{(t,f)\in S} \log\left(1 - \gamma_{t,f}\right) < \log(1 - r_0), \tag{6}$$

where $\mathcal{P}$ is the set of all the paths from the first node to the last node of the scheduling graph. The problem described by Eqn.(5)-(6) corresponds to a *constrained shortest path* (CSP) problem.

3) Algorithm Selection: The LARAC Algorithm: A plethora of algorithms have been developed to solve the CSP problem [54]. However, these algorithms usually assume that the constrained metric is positive and construct the result path hop-by-hop, thereby starting with feasible paths (i.e., paths satisfying Eqn. (6)) and by avoiding reaching infeasible paths (i.e., paths not satisfying Eqn. (6)) when defining the subsequent hops. However, our constrained metric is negative (because $\gamma_{t,f} < 1$). As such, path construction starts from infeasible paths and has to end in the feasible zone. Hop-by-hop algorithms are not straightforward to adapt to this situation. The *LARAC* algorithm [60]–[63], one of the best performing CSP algorithms [54], does not construct its result path hop-by-hop. Instead, it computes subsequent paths using Dijkstra with an aggregate cost metric and labels the intermediate paths found as either infeasible or feasible. Because of this (and because Dijkstra can handle loopless graphs with negative metrics), the LARAC algorithm can be directly used to solve problem (5)-(6).

4) Integration of the Scheduler in the WDetServ Model: In this section, we describe the integration of the proposed reliability-aware scheduler with the WDetServ model presented in Sec. II-C and its interaction with the interface.

To check if an application request can be accommodated with a given target reliability level (HASACCESS arrow in Fig. 3), the scheduler runs the LARAC algorithm on the scheduling graph. If the algorithm returns a path, there is a feasible schedule and access is granted. Otherwise, the request is denied. Once an application has to be scheduled (resp. unscheduled) (*schedule* (resp. *unschedule*) arrow in Fig. 3), the scheduler removes (resp. adds back) the edges constituting the found (resp. previously allocated) path from the scheduling graph, thereby ensuring that other applications cannot (resp. can) use the reserved (resp. freed) RBs. Once an application has to be rescheduled (*reschedule* arrow in Fig. 3), the scheduler simply unschedules it and then schedules it again. In case no schedule (i.e., no path) is found, the scheduler notifies the upper layers that the rescheduling failed. In this way, for instance, the application can be served by other network interfaces. If applications have different priorities, the highest priority applications are rescheduled first, such that the applications which have to be evicted from the network are the least priority ones. Upon reception of updated channels statistics (*channel stats* arrow in Fig. 3), the scheduler updates the $\gamma_{t,f}$ values associated to the different edges of its scheduling graph.

In order to handle sub-frames, the scheduler still operates identically and on the same scheduling graph. However, when running the LARAC algorithm, the scheduling graph is split according to the considered sub-frame size. In the example of Fig. 4, for a sub-frame size of two slots, the scheduling graph would be split at node n_2. For a sub-frame size of one slot, the scheduling graph would be split at nodes n_1, n_2 and n_3. The scheduler then runs the LARAC algorithm on the different sub-graphs and returns a schedule only if a path has been found on *all* the sub-graphs. This ensures that the target reliability level

---

[1] We assume that $\gamma_{t,f} < 1$, which allows the usage of the log operator.

is guaranteed for all the iterations of the considered sub-frame within the super-frame.

IV. IMPLEMENTATION DETAILS

In this section the details of the implementation of the dynamic scheduling procedure in a TSCH industrial WSN are presented. Dynamic scheduling is a periodic operation which consists in *(i)* sending up-to-date channel quality information to the scheduler, and *(ii)* retrieving up-to-date schedules from the scheduler, in a timely manner.

In our implementation, data communication between the motes and the WG (introduced in Sec. II-C1) is always uplink. For every RB allocated by the schedule, a given mote transmits either a data packet, if available, or a substitute probing packet. The WG receives the whole traffic of the network and it is assumed to be capable of selectively receiving or transmitting over the total 16 IEEE 802.15.4 frequencies of the 2.4GHz ISM unlicensed band. Normal motes are limited to a single frequency for transmission or reception in every time slot.

At the end of every frame, the WG sends the up-to-date link qualities to the scheduler, which replies with new schedules fulfilling the target reliability according to the new link qualities. In this way, at the beginning of the next frame, the WG can distribute the new schedules by including them in the Enhanced Beacon (EB), a broadcast control message which provides necessary information for the operation of the TSCH network, such as synchronization and radio resource management. It can happen that this message is not correctly received by all the devices. In this case, the motes can temporarily silence their radios (avoiding potential interference in the network) and wait for the next EB. As control messages can be transmitted using robust radio resources (e.g. non-interfered frequencies), in our evaluation, for simplicity, we assume that no EB is lost and we leave the evaluation of the network under this condition as future work.

The details of channel quality estimation procedure are described in Sec. IV-A, while Sec. IV-B and IV-C present the different types of schedulers implemented for the evaluation.

A. Channel Quality Estimation

In our implementation, channel quality estimation is passively performed by the WG upon reception of data packets. For every mote, the WG estimates the PDR of every frequency used for communication over time. The results of the estimation are sent to the scheduler at the end of every super-frame, which keeps the network radio resources up-to-date.

Extending Eqn. (III), we define the PDR of a resource block corresponding to time-frequency pair (t, f) for a specific mote m as $\mathrm{RB}_{t,f}^{m}$. The PDR exhibits a stochastic behavior that can be modeled by a Bernoulli random variable [50]

$$\mathrm{RB}_{t,f}^{m} \sim \mathrm{Bern}\left(\gamma_{t,f}^{m}\right) \qquad \mathrm{E}\left[\mathrm{RB}_{t,f}\right] = \gamma_{t,f}^{m}. \tag{7}$$

Its expected value $\gamma_{t,f}^{m}$ is the PDR of the frequency f at time t of mote m. For every RB in the schedule S^m of mote m, the WG updates the corresponding PDR with the outcome of the transmission by feeding an Exponentially Weighted Moving Average (EWMA) filter.

$$\gamma_{t+1,f}^{m} = \begin{cases} \alpha + (1-\alpha)\,\gamma_{t,f}^{m} & \text{if } \mathrm{RB}_{t,f}^{m} = 1 \\ (1-\alpha)\,\gamma_{t,f}^{m} & \text{if } \mathrm{RB}_{t,f}^{m} = 0 \end{cases}, \; (t,f) \in S^m. \tag{8}$$

Here, $\alpha \in [0,1]$ represents the coefficient of weighting decrease, and can be tuned to select the best trade-off between reactivity and stability. The EWMA filter has proven to be a valid design choice for channel quality estimation in WSN as it is able to quickly adapt to the highly varying dynamic of these channels [64], [65].

Furthermore, we introduce an aging process for all the unscheduled frequencies in a specific time slot. It consists of artificially improving the PDR of unscheduled frequencies such that they can be explored in the future and not blacklisted by the scheduler. The aging process is implemented using the EWMA filter. For every transmission opportunity, the PDRs of all the resource blocks are artificially "improved" by

$$\gamma_{t+1,f}^{m} = \beta + (1-\beta)\,\gamma_{t,f}^{m}\,,\; (t,f) \notin S^m. \tag{9}$$

The value of β must be chosen carefully as it introduces a potential penalty in reliability when exploring channels with low PDR, and should be, as discussed in Sec. V-C, tailored to the interference behavior.

B. Reliability-Based Scheduler Cost Function

As a cost function, we use the exemplary function described in Sec. III-2. The cost function defines $c_s = 0$ as the cost of silent edges and $c_{t,f} = 1$ as the cost of all the other edges. This function leads to the assignment of the minimum amount of RBs to each application. The evaluation of the impact of the cost function on the performance of the scheduler is left for future work.

C. Comparing to State-of-the-Art Schedulers

As there are no schedulers addressing the reliability in our scenario, we compare our approach with the state-of-the-art scheduling routines. We have selected a classic approach

Algorithm 1 MaximumThroughputScheduler

```
Set of motes, availabilities $m \in \mathcal{M}, r^m \leftarrow 0$
$\mathcal{A}$                                              ▷ resulting allocation
for $t$ in $s_0$ do                                        ▷ all slots
    $\mathcal{M}_t \leftarrow \mathcal{M}$; $\mathcal{A}_t \leftarrow \emptyset$          ▷ frequency allocations
    do
        for $f$ in $\mathcal{F}_t$ do                       ▷ all frequencies
            $\mathcal{S}_f \leftarrow \emptyset$                        ▷ set of scores
            for $m \in \mathcal{M}_t$ do
                $\mathcal{S}_f \leftarrow sc_f^m = \gamma_{t,f}^m$
            $\mathcal{A}_t \leftarrow (m^\star, f) = \arg\max_m(\mathcal{S}_f)$
        for all $(m^\star, f) \in \mathcal{A}_t$ do
            if count($m^\star$ in $\mathcal{A}_t$)$> 1$ then    ▷ Enforce 1 freq./mote
                $f^\star = \arg\max_{\forall (m^\star, f) \in \mathcal{A}_t}(sc_f^{m^\star})$
            else $f^\star = f$
            $\mathcal{A} \leftarrow (m^\star, f^\star, t)$                  ▷ allocate
            $\mathcal{F}_t = \mathcal{F}_t - f^\star$; $\mathcal{M}_t = \mathcal{M}_t - m^\star$
            $r^{m^\star} = r^{m^\star} + \gamma_{t,f^\star}^{m^\star}$
            if $r^{m^\star} \geq r_0$ then $\mathcal{M} \leftarrow \mathcal{M} - m^\star$
    while $\mathcal{F}_t \neq \emptyset \wedge \mathcal{M}_t \neq \emptyset$       ▷ repeat if freq. left
return $\mathcal{A}$
```

known from the Long Term Evolution (LTE) resource scheduling: maximum throughput (MT). MT scheduler is a greedy approach which allocates the resources to the applications which can make the best use of them. In the case of LTE, this implies users with higher channel quality. In our case, however, the throughput (per RB) is equivalent to the PDR $\gamma_{t,f}^{m}$. Hence, we allocate a RB to the application with the highest success probability. The basic pseudocode for the scheduler is presented in Alg. 1. The original MT scheduler is classified as best-effort. Our adapted version has a certain degree of QoS-awareness. If a mote has already achieved its reliability requirement, it is not allocated any more RBs (Alg. 1, line 18).

V. RELIABILITY-BASED SCHEDULING EVALUATION

In this section, the reliability-based wireless scheduler presented in Sec. III is evaluated by means of network-wide simulations of an IWSN operating in an interfering environment. For this, a custom Python discrete event simulator is used.

First, Sec. V-A summarizes the details of our simulations, together with the wireless propagation environment and the model of the external interference. Then, in Sec. V-B, the simulation results of our reliability-based scheduler are shown and discussed in different interfering scenarios. Finally, in Sec. V-C, we discuss the observations from the evaluation.

A. Simulation Details

The evaluation is performed for a star-topology WSN network where all the motes are placed at the same distance $d = 10$ m from the WG and use the same transmission power equal to 20 dBm. The super-frame consists of $s_0 = 8$ slots and every slot has a duration of $t_s = 10$ ms. The application periodically generates traffic every 8 ms.

Every simulation performs the following steps. Initially, in the absence of channel quality information, an arbitrary default PDR is assumed for every frequency. Motes join the network until the scheduler does not find a schedule that fulfills the required reliability. Afterwards, actual packet transmission starts, and correct channel quality information is provided to the scheduler, which adjusts the number of devices in the network, eventually de-allocating devices if the reliability level is not reached and not enough RBs are left in the system.

Our industrial WSN network operates in an indoor environment in the 2.4 GHz ISM unlicensed band and it is subject to external interference from coexisting wireless technologies. In the remaining part of this section the wireless propagation model and the interference model are described in details.

1) Wireless Propagation Model: In our simulations, a stochastic wireless channel model for WSNs is adopted. As shown in [66], every transmission experiences a channel realization obtained by adding path loss and multi-path fading, capturing the attenuation and amount of reflection of a particular environment, $P_{\text{PL}} = (c/(4\pi f d))^{\eta}$, $\ln P_{\text{FF}} \sim \mathcal{N}\left(0, \sigma^2\right)$.

The path loss P_{PL} depends on the receiver distance d, the carrier frequency f, the speed of light c and the path loss exponent η. The effect of multi-path is modeled by a log-normal distribution, suitable for short-range indoor propagation [66] and depends on σ^2, the amount of multi-path fading in the environment. In our simulations, $\sigma^2 = 6.1$ dB and $\eta = 2.23$.

2) External Interference Model: In the simulations we take external Wi-Fi interference into account. This might arise from coexisting Wi-Fi networks used in the factory. The different Wi-Fi networks simultaneously operate in the 2.4 GHz ISM band. Common Wi-Fi deployments allow up to three simultaneous non-overlapping Access Points (AP) on Wi-Fi channels 1, 6, 11. Interfering Wi-Fi APs are co-located with the WSN devices and, in our simulations, may vary their transmission power selecting different power levels: -20, -10, 0, 10 dBm. The traffic pattern of Wi-Fi clients and APs is approximated using a Bernoulli random variable with parameter p, with 1 denoting presence of interference and 0 absence of it. We simulate the Wi-Fi interference as in [50], distributing the received power from every AP over multiple WSN channels.

B. Simulation Results

In this section, the simulation results of the proposed schedulers together with the channel quality estimator are comprehensively evaluated in different interference scenarios.

First, in Sec. V-B1, the correctness of the reliability-based scheduler of Sec. III is evaluated with simulations under the assumption of perfect channel quality information. Second, in Sec. V-B2, the impact of channel quality estimation is evaluated in a static interference scenario. In Sec. V-B3, dynamic interference scenarios are shown and the suitability of our dynamic scheduling algorithm is demonstrated. Finally, in Sec. V-B4, we present the performance of the MT scheduler, and compare it to the reliability-based scheduler.

1) Perfect Channel Quality Information: Simulations under the assumption of perfect channel quality information are performed to verify the correctness and test the performance of the reliability-based scheduling algorithm independently from the channel quality estimation. Fig. 5 and 6 show the empirical cumulative distribution function of the packet delay and reliability for all the motes supported by the scheduler during the entire simulation. The results show that the proposed scheduler is able to fulfill target reliabilities of 0.9 and 0.99, and fulfills the latency bound of 160 ms by achieving a maximum delay of 80 ms when the channel quality information is known.

An exemplary schedule computed to support a target reliability of 0.9 is $\{(0,9),(2,11),(6,11)\}$. It allocates 3 re-transmissions in the super-frame at time instants $0,2,6$ over the frequencies $9,11$ with corresponding PDR equal to $0.68, 0.9$ respectively. The allocation provides an application reliability of 0.987, which fulfills the requirement of 0.9.

2) Online Channel Quality Estimation: Here, we evaluate the impact of different EWMA parameters on the results of the scheduling algorithm in a static interference scenario.

Table I summarizes the results for different values of α, for a duration of 16.7 minutes. For lower values of α the filter includes more history in the estimation, thus adapting slowly to the changes. On the other hand, for higher values of α the filter is more reactive. The extremes of this dynamic can be observed in Table I. When α has the values $0.001, 0.1$, the impact of erroneous estimation is fatal and the scheduler cannot meet the average reliability requirements. Moreover, for the remaining values of α, although the target reliability is

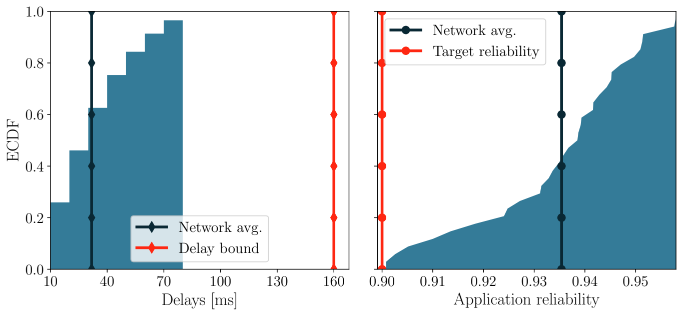

Fig. 5: Reliability-based scheduler. 0.90 reliability, 34 motes, 16.7 minutes.

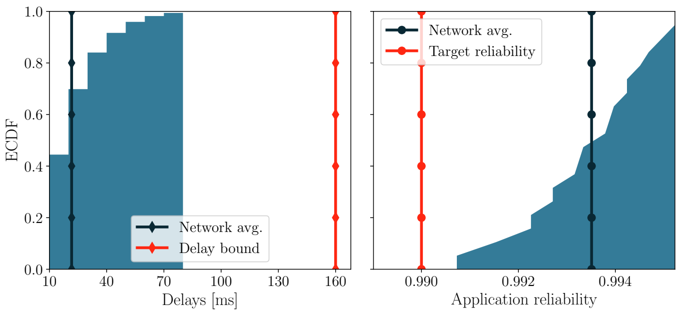

Fig. 6: Reliability-based scheduler. 0.99 reliability, 19 motes, 16.7 minutes.

α	Network reliability: avg. / tar. (% motes)	Network avg. delay	Motes
0.001	0.872 / 0.9 (12.5)	35.38	40
0.01	0.919 / 0.9 (97)	32.78	34
0.02	0.920 / 0.9 (94)	32.89	35
0.03	**0.924 / 0.9 (100)**	**32.70**	**35**
0.04	0.921 / 0.9 (97)	32.75	34
0.05	0.920 / 0.9 (94)	32.80	35
0.06	0.915 / 0.9 (89)	33.24	35
0.1	0.891 / 0.9 (74)	33.77	34

Tab. I: Evaluation of the EWMA weighting factor α in the online LQE scenario, $\beta = 10^{-6}$. The avg. network reliability is compared with the target reliability (0.9). "%" indicates the percentage of motes achieving the target reliability. The last two columns show the average network delay and the number of served motes.

achieved by the network, we can observe that the totality of the motes does not achieve the target reliability. This effect is due to the interplay between estimation error and the number of transmitted packets and it is discussed in Sec. V-C. However, for the specific value $\alpha = 0.03$, both the network average reliability and the per-mote average reliability are achieved.

The details of the simulation when $\alpha = 0.03$ are shown in Fig. 7. On the left, the time evolution of the average network channel quality, together with the number of supported motes supported, are shown. At the end of the simulation our system is able to support 35 devices fulfilling their QoS requirements. In this case, the combination between α and the number of transmitted packets yields to a successful channel quality estimation. As discussed in Sec. V-C, this motivates further work where the interplay between estimation errors and the scheduler can be studied.

3) Dynamic interference: Finally, it is interesting to observe the performance of the scheduler in presence of dynamic interference. Also in this case, results are shown for $\alpha = 0.03$, which leads to a good trade-off between stability and reactivity of the channel quality estimator for our scheduler. Fig. 8 shows the performance of the scheduler in presence of increasing Wi-Fi transmission power $\mathrm{P_{tx}}$. The transmission power is increased during the simulation in four steps from -20 dBm to 10 dBm. Also in this case, the system is able to detect the interference and adapt the radio resources guaranteeing the target reliability and delay bound. In this case, due to higher interference, the number of supported devices in the network is 22.

4) Comparing to the Maximum Throughput Scheduler: In Fig. 9, we plot the evaluation results of the MT scheduler implementing the algorithm from Sec. IV-C in the dynamic interference scenario, with the same simulation parameters as the proposed reliability-based scheduler in Fig. 8. We observe that the MT scheduler achieves the target reliability for all motes, with an average of 0.917. The delay ECDF of the MT scheduler is comparable to the reliability-based scheduler. As MT is greedy with respect to the frequency allocations, it consumes more resources, which has resulted in slightly higher average reliability at the expense of supporting only 9 motes, compared to the reliability-based scheduler which supports 22.

C. *Results Discussion*

The simulations prove that the proposed reliability-based dynamic scheduler is able to provide the required industrial QoS in presence of dynamic interference. In this work, proof-of-concept results demonstrate the suitability of the reliability-based scheduler for IWSN, and leave space for further improvements which are discussed in this section.

The performance of the system highly depends on *(i)* the precision of the channel quality estimator and *(ii)* the number of packets transmitted in a specific scenario. In fact, errors in channel quality estimation can lead to erroneous schedules, and few transmitted packets lead to an insufficient number of samples to observe the *average* reliability computed by the scheduler. The investigation of different link estimation techniques on the proposed scheduler is left as future work.

Additionally, two effects introduce a penalty on the system performance and should be improved in further work: *(i)* the channel quality estimation is currently performed using data packets; *(ii)* the aging process adopted to explore new links introduces capacity in the system that is explored using precious data packets. Instead, specific *probing packets* should be sent to estimate the quality of the communication channel. This prevents packet loss when the channel quality worsens, therefore improving the reliability of the application. Thus, in order to improve the system performance, a probing mechanism should be designed and introduced.

Finally, we analyze the dynamics of the scheduler with increasing interference conditions. This leads to critical situations where some devices cannot be served by the network. This is done to evaluate the performance of the scheduler, however, in a real implementation, the devices should be served by backup links, and the effect of interference should be carefully taken into account such that the QoS can be guaranteed even in worst-case conditions. Also in this case, a probing procedure is needed to acquire accurate channel quality information.

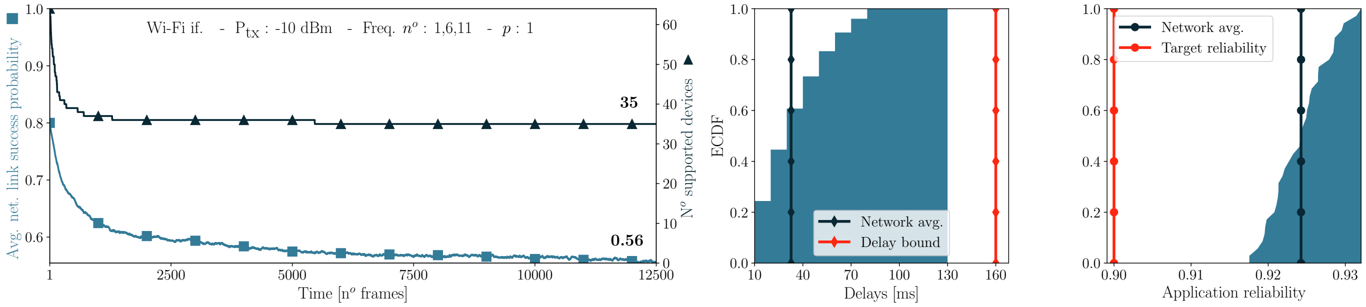

Fig. 7: Reliability-based scheduler. Static interference, $\alpha = 0.03$, $\beta = 10^{-6}$, 35 motes, 16.7 minutes.

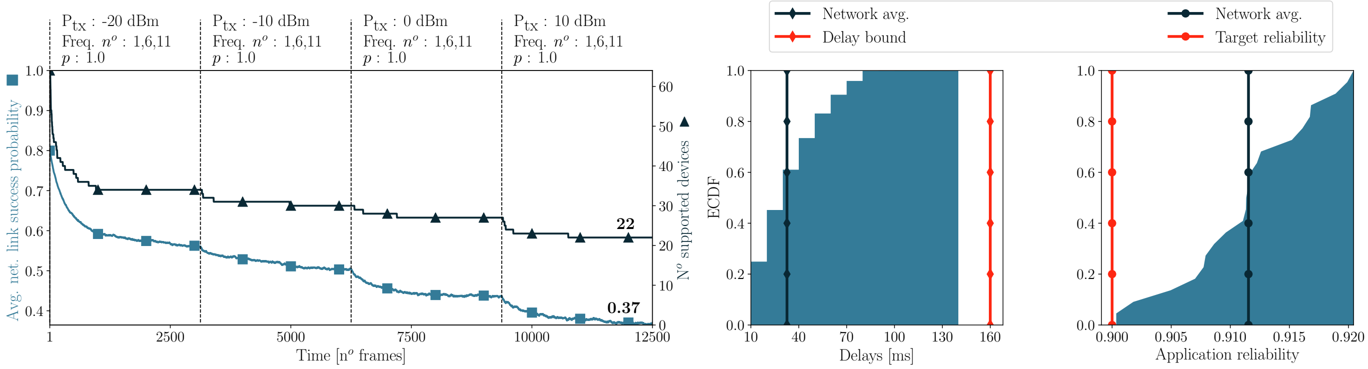

Fig. 8: Reliability-based scheduler. Increasing Wi-Fi transmission power (P_{tx}), $\alpha = 0.03$, $\beta = 10^{-6}$, 22 motes, 16.7 minutes.

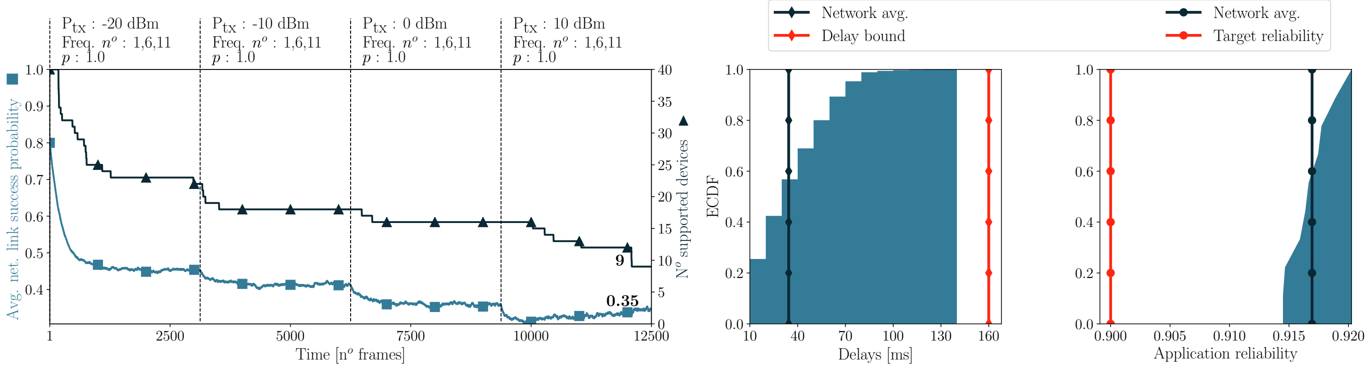

Fig. 9: Maximum throughput (MT) scheduler. Increasing Wi-Fi transmission power (P_{tx}), $\alpha = 0.04$, $\beta = 10^{-6}$, 9 motes, 16.7 minutes.

VI. CONCLUSIONS AND FUTURE WORK

In this article we solve the issue of QoS provisioning in hybrid wired/wireless networks for industrial applications. We achieve this by providing an interface that enables seamless industrial QoS provisioning across wired and wireless domains. In order to achieve this, the *delay bound* and *target reliability* of an application must be fulfilled in both networks.

By extending the work in [4] with the WDetServ model, we provide the first framework for seamless QoS provisioning in hybrid networks. The model introduces a worst-case delay calculation and a reliability-based scheduler, together able to provide industrial QoS in IWSNs.

However, in IWSN, guaranteeing reliability is a difficult task. To tackle this challenge, we propose the first reliability-based dynamic scheduler for TSCH which is able to guarantee a *target reliability* in presence of dynamic interference, and is needed to enable seamless QoS provisioning in hybrid networks. The scheduler provides reliability combining retransmissions over interfered channels, thus efficiently using all available radio resources. The proposed scheduler was evaluated in a proof-of-concept IWSN in presence of dynamic interference. The results prove the suitability and flexibility of the proposed scheduler and motivate further work. In particular, the interplay between the scheduler and the link quality estimation should be investigated, and a new probing mechanism should be designed to retrieve link quality information in a timely manner and ensure the safe operation of the network.

REFERENCES

[1] R. Zurawski, *Industrial Communication Technology Handbook*. CRC Press, 2014.

[2] P. Gaj, J. Jasperneite, and M. Felser, "Computer Communication within Industrial Distributed Environment – A Survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 182–189, 2013.

[3] J. W. Guck, M. Reisslein, and W. Kellerer, "Model-Based Control Plane for Fast Routing in Industrial QoS Network," in *23rd International Symposium on Quality of Service (IWQoS)*. IEEE, 2015, pp. 65–66.

[4] ——, "Function Split Between Delay-Constrained Routing and Resource Allocation for Centrally Managed QoS in Industrial Networks," *IEEE*

Transactions on Industrial Informatics, vol. 12, no. 6, pp. 2050–2061, Dec 2016.

[5] J. W. Guck, A. V. Bemten, and W. Kellerer, "DetServ: Network Models for Real-Time QoS Provisioning in SDN-Based Industrial Environments," *IEEE Transactions on Network and Service Management*, vol. 14, no. 4, pp. 1003–1017, Dec 2017.

[6] R. Tavakoli, M. Nabi, T. Basten, and K. Goossens, "Enhanced Time-Slotted Channel Hopping in WSNs Using Non-Intrusive Channel-Quality Estimation," in *IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, 2015, pp. 217–225.

[7] L. Seno, S. Vitturi, and F. Tramarin, "Experimental Evaluation of the Service Time for Industrial Hybrid (Wired/Wireless) Networks Under Non-Ideal Environmental Conditions," in *IEEE 16th Conference on Emerging Technologies & Factory Automation (ETFA)*, 2011, pp. 1–8.

[8] S. Knauth, R. Kistler, D. Kaslin, and A. Klapproth, "SARBAU Towards Highly Self-Configuring IP-Fieldbus Based Building Automation Networks," in *22nd International Conference on Advanced Information Networking and Applications (AINA)*. IEEE, 2008, pp. 713–717.

[9] 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 Transactions on Industrial Electronics*, vol. 49, no. 3, pp. 693–701, 2002.

[10] D. Miorandi and S. Vitturi, "Hybrid Wired/Wireless Implementations of Profibus DP: A Feasibility Study Based on Ethernet and Bluetooth," *Computer Communications*, vol. 27, no. 10, pp. 946–960, 2004.

[11] T. Zhong, Z. Peng, Y. Haibin, and W. Hong, "Zigbee-Based Wireless Extension of Foundation Fieldbus," in *6th IEEE International Conference on Industrial Informatics (INDIN).*, 2008, pp. 661–666.

[12] S. Trikaliotis and A. Gnad, "Mapping WirelessHart into Profinet and Profibus Fieldbusses," in *IEEE Conference on Emerging Technologies & Factory Automation (ETFA)*, 2009, pp. 1–4.

[13] S. Vitturi, I. Carreras, D. Miorandi, L. Schenato, and A. Sona, "Experimental Evaluation of an Industrial Application Layer Protocol Over Wireless Systems," *IEEE Transactions on Industrial Informatics*, vol. 3, no. 4, pp. 275–288, 2007.

[14] A. Flammini, D. Marioli, E. Sisinni, and A. Taroni, "Design and Implementation of a Wireless Fieldbus for Plastic Machineries," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 3, pp. 747–755, 2009.

[15] A. Flammini, P. Ferrari, D. Marioli, E. Sisinni, and A. Taroni, "Wired and Wireless Sensor Networks for Industrial Applications," *Microelectronics journal*, vol. 40, no. 9, pp. 1322–1336, 2009.

[16] G. Cena, L. Seno, A. Valenzano, and C. Zunino, "On the Performance of IEEE 802.11e Wireless Infrastructures for Soft-Real-Time Industrial Applications," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 3, pp. 425–437, 2010.

[17] J. Frey, A. Kreitz, and G. Scheibe, "Unplugged but Connected," *Elektron Journal, South African Institute of Electrical Engineers*, vol. 23, no. 3, p. 44, 2006.

[18] T. Sauter, "The Three Generations of Field-Level Networks - Evolution and Compatibility Issues," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, pp. 3585–3595, 2010.

[19] G. Cena, A. Valenzano, and S. Vitturi, "Hybrid Wired/Wireless Networks for Real-Time Communications," *IEEE Industrial Electronics Magazine*, vol. 2, no. 1, 2008.

[20] D. Dujovne, T. Watteyne, X. Vilajosana, and P. Thubert, "6TiSCH: Deterministic IP-Enabled Industrial Internet (of Things)," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 36–41, 2014.

[21] P. Thubert, M. R. Palattella, and T. Engel, "6TiSCH Centralized Scheduling: When SDN meet IoT," in *IEEE Conference on Standards for Communications and Networking (CSCN)*, 2015, pp. 42–47.

[22] Y. Yuan, Z. Yang, Z. He, and J. He, "An Integrated Energy Aware Wireless Transmission system for QoS Provisioning in Wireless Sensor Network," *Computer Communications*, 2006.

[23] X. Zhang, J. Tang, H.-H. Chen, S. Ci, and M. Guizani, "Cross-Layer-Based Modeling for Quality of Service Guarantees in Mobile Wireless Networks," *IEEE Communications Magazine*, vol. 44, no. 1, pp. 100–106, Jan 2006.

[24] P. Mohapatra, J. Li, and C. Gui, "Qos in mobile ad hoc networks," *IEEE Wireless Communications*, vol. 10, no. 3, pp. 44–53, 2003.

[25] A. Malik, J. Qadir, B. Ahmad, K.-L. A. Yau, and U. Ullah, "QoS in IEEE 802.11-Based Wireless Networks: A Contemporary Review," *Journal of Network and Computer Applications*, vol. 55, pp. 24–46, 2015.

[26] J. Blanckenstein, C. Nardin, J. Klaue, and H. Karl, "Error Characterization of Multi-Access Point WSNs in an Aircraft Cabin," in *IEEE International Conference on Communication Workshop (ICCW)*, June 2015, pp. 2363–2368.

[27] G. Cena, S. Scanzio, and A. Valenzano, "Seamless Link-Level Redundancy to Improve Reliability of Industrial Wi-Fi Networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 2, pp. 608–620, April 2016.

[28] G. Song and Y. Li, "Cross-Layer Optimization for OFDM Wireless Networks-Part II: Algorithm Development," *IEEE Transactions on Wireless Communications*, vol. 4, no. 2, pp. 625–634, 2005.

[29] N. Abu-Ali, A.-E. M. Taha, M. Salah, and H. Hassanein, "Uplink Scheduling in LTE and LTE-Advanced: Tutorial, Survey and Evaluation Framework," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1239–1265, 2014.

[30] M. A. Mehaseb, Y. Gadallah, A. Elhamy, and H. Elhennawy, "Classification of LTE Uplink Scheduling Techniques: An M2M Perspective," *IEEE Communications Surveys Tutorials*, vol. 18, no. 2, pp. 1310–1335, Secondquarter 2016.

[31] F. Capozzi, G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, "Downlink Packet Scheduling in LTE Cellular Networks: Key Design Issues and a Survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 678–700, 2013.

[32] N. Accettura, M. R. Palattella, G. Boggia, L. A. Grieco, and M. Dohler, "DeTAS: A Decentralized Traffic Aware Scheduling Technique Enabling IoT-Compliant Multi-Hop Low-Power and Lossy Networks," in *Second IEEE WoWMoM Workshop on the Internet of Things: Smart Objects and Services, IoT-SoS*, vol. 26, 2013.

[33] N. Accettura, E. Vogli, M. R. Palattella, L. A. Grieco, G. Boggia, and M. Dohler, "Decentralized Traffic Aware Scheduling in 6TiSCH Networks: Design and Experimental Evaluation," *IEEE Internet of Things Journal*, vol. 2, no. 6, pp. 455–470, 2015.

[34] A. Hannachi and A. Bachir, "Distributed Cell Scheduling for Multichannel IoT MAC Protocols," in *13th International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, 2017.

[35] R. Soua, P. Minet, and E. Livolant, "DiSCA: A Distributed Scheduling for Convergecast in Multichannel Wireless Sensor Networks," in *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2015, pp. 156–164.

[36] ——, "A Distributed Joint Channel and Slot Assignment for Convergecast in Wireless Sensor Networks," in *6th International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 2014.

[37] ——, "Wave: a Distributed Scheduling Algorithm for Convergecast in IEEE 802.15.4e TSCH Networks," *Transactions on Emerging Telecommunications Technologies*, vol. 27, no. 4, pp. 557–575, 2016.

[38] M. R. Palattella, T. Watteyne, Q. Wang, K. Muraoka, N. Accettura, D. Dujovne, L. A. Grieco, and T. Engel, "On-The-Fly Bandwidth Reservation for 6TiSCH Wireless Industrial Networks," *IEEE Sensors Journal*, vol. 16, no. 2, pp. 550–560, 2016.

[39] E. Municio and S. Latré, "Decentralized Broadcast-Based Scheduling for Dense Multi-Hop TSCH Networks," in *Proceedings of the Workshop on Mobility in the Evolving Internet Architecture*. ACM, 2016, pp. 19–24.

[40] A. Morell, X. Vilajosana, J. L. Vicario, and T. Watteyne, "Label Switching over IEEE 802.15.4e Networks," *Transactions on Emerging Telecommunications Technologies*, vol. 24, no. 5, pp. 458–475, 2013.

[41] M. Vilgelm, M. Gürsu, S. Zoppi, and W. Kellerer, "Time Slotted Channel Hopping for Smart Metering: Measurements and Analysis of Medium Access," in *IEEE International Conference on Smart Grid Communications*, 2016, pp. 109–115.

[42] M. R. Palattella, N. Accettura, M. Dohler, L. A. Grieco, and G. Boggia, "Traffic Aware Scheduling Algorithm for Reliable Low-Power Multi-Hop IEEE 802.15.4e Networks," in *IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2012.

[43] M. R. Palattella, N. Accettura, L. A. Grieco, G. Boggia, M. Dohler, and T. Engel, "On Optimal Scheduling in Duty-Cycled Industrial IoT Applications Using IEEE802.15.4e TSCH," *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3655–3666, 2013.

[44] R. Soua, P. Minet, and E. Livolant, "MODESA: an Optimized Multichannel Slot Assignment for Raw Data Convergecast in Wireless Sensor Networks," in *IEEE 31st International Performance Computing and Communications Conference (IPCCC)*, 2012, pp. 91–100.

[45] R. Soua, E. Livolant, and P. Minet, "MUSIKA: A Multichannel Multi-Sink Data Gathering Algorithm in Wireless Sensor Networks," in *9th International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, 2013, pp. 1370–1375.

[46] Á. A. Farías and D. Dujovne, "A Queue-Based Scheduling Algorithm for PCE-Enabled Industrial Internet of Things Networks," in *Sixth Argentine Conference on Embedded Systems (CASE)*. IEEE, 2015, pp. 31–36.

[47] Y. Jin, P. Kulkarni, J. Wilcox, and M. Sooriyabandara, "A Centralized Scheduling Algorithm for IEEE 802.15.4e TSCH Based Industrial Low

Power Wireless Networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2016, pp. 1–6.

[48] T.-S. Chen, S.-Y. Kuo, and C.-H. Kuo, "Scheduling for Data Collection in Multi-hop IEEE 802.15. 4e TSCH Networks," in *International Conference on Networking and Network Applications*. IEEE, 2016.

[49] G. Patti and L. L. Bello, "A Priority-Aware Multichannel Adaptive Framework for the IEEE 802.15.4e-LLDN," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 10, pp. 6360–6370, 2016.

[50] S. Zoppi, H. M. Gürsu, M. Vilgelm, and W. Kellerer, "Reliable Hopping Sequence Design for Highly Interfered Wireless Sensor Networks," in *IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, 2017, pp. 1–7.

[51] M. Gürsu, M. Vilgelm, S. Zoppi, and W. Kellerer, "Reliable Co-Existence of 802.15.4e TSCH-Based WSN and Wi-Fi in an Aircraft Cabin," in *IEEE International Conference on Communications Workshops*, May 2016, pp. 663–668.

[52] L. Seno, F. Tramarin, and S. Vitturi, "Performance of Industrial Communication Systems: Real Application Contexts," *IEEE Industrial Electronics Magazine*, vol. 6, no. 2, pp. 27–37, 2012.

[53] "Communication delivery time performance requirements for electric power substation automation," *IEEE Std 1646-2004*, pp. 1–24, 2005.

[54] J. W. Guck, A. Van Bemten, M. Reisslein, and W. Kellerer, "Unicast QoS Routing Algorithms for SDN: A Comprehensive Survey and Performance Evaluation," *IEEE Communications Surveys & Tutorials*, 2017.

[55] J.-Y. L. Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*, April 2012. [Online]. Available: http://ica1www.epfl.ch/PS_files/netCalBookv4.pdf

[56] A. Van Bemten and W. Kellerer, "Network Calculus: A Comprehensive Guide," *Technical University of Munich, Chair of Communication Networks, Technical Report No. 201603*, October 2016.

[57] R. Steigmann and J. Endresen, "Introduction to WISA: WISA-Wireless Interface for Sensors and Actuators," *White paper, ABB*, 2006.

[58] J. Song, S. Han, A. Mok, D. Chen, M. Lucas, M. Nixon, and W. Pratt, "WirelessHART: Applying Wireless Technology in Real-Time Industrial Process Control," in *IEEE Real-Time and Embedded Technology and Applications Symposium.*, 2008, pp. 377–386.

[59] T. Watteyne, M. Palattella, and L. Grieco, "Using IEEE 802.15.4e Time-Slotted Channel Hopping (TSCH) in the Internet of Things (IoT): Problem Statement," RFC Editor, RFC 7554, May 2015.

[60] Y. P. Aneja and K. Nair, "The Constrained Shortest Path Problem," *Naval Research Logistics Quarterly*, vol. 25, no. 3, pp. 549–555, 1978.

[61] G. Y. Handler and I. Zang, "A Dual Algorithm for the Constrained Shortest Path Problem," *Networks*, vol. 10, no. 4, pp. 293–309, 1980.

[62] D. Blokh and G. Gutin, "An Approximate Algorithm for Combinatorial Optimization Problems with Two Parameters," *Australasian Journal of Combinatorics*, vol. 14, pp. 157–164, 1996.

[63] A. Jüttner, B. Szviatovski, I. Mécs, and Z. Rajkó, "Lagrange Relaxation Based Method for the QoS Routing Problem," in *Proceedings of IEEE INFOCOM*, vol. 2, 2001, pp. 859–868.

[64] A. Woo and D. E. Culler, *Evaluation of Efficient Link Reliability Estimators for Low-Power Wireless Networks*. Computer Science Division, University of California Oakland, Calif, USA, 2003.

[65] N. Baccour, A. Koubâa, L. Mottola, M. A. Zúñiga, H. Youssef, C. A. Boano, and M. Alves, "Radio Link Quality Estimation in Wireless Sensor Networks: A Survey," *ACM Transactions on Sensor Networks (TOSN)*, vol. 8, no. 4, p. 34, 2012.

[66] I. F. Akyildiz and M. C. Vuran, *Wireless Sensor Networks*. John Wiley & Sons, 2010, vol. 4.

Samuele Zoppi (S'16) was born in Turin, Italy, in 1992. He received the M.Sc. degree in Telecommunications Engineering from the Polytechnic University of Turin in October 2015. In January 2016, he joined the Chair of Communication Networks with the Department of Electrical and Computer Engineering at the Technical University of Munich, where he is currently pursuing his Ph.D. degree in Communication Networks. His current research interest is dynamic radio resource management of industrial Wireless Sensor Networks for Networked Control Systems.

Amaury Van Bemten (S'18) was born in Liège, Belgium, in 1993. He received the B.Sc. degree in Engineering in June 2013 and the M.Sc. in Computer Science and Engineering in June 2015, both from the University of Liège (Belgium). In September 2015, he joined the Chair of Communication Networks at the Technical University of Munich (TUM), where he is currently pursuing the Ph.D. degree as a member of the research and teaching staff. His current research focuses on routing algorithms and the application of software-defined networking for resilient real-time communications in industrial environments.

H. Murat Gürsu (S'16) was born in Istanbul, Turkey in 1989. He received the BSc. degree in Electrical and Electronics Engineering from Bogazici University in 2012. He received the MSc. in Communication Engineering in Technical University of Munich (TUM) in 2014. Currently he is working towards his Ph.D. degree at the TUM, Chair of Communication Networks in the Department of Electrical and Computer Engineering since 2015. His research interests include wireless resource management and wireless sensor networks.

Mikhail Vilgelm (S'16) received the Diplom-Engineer degree from the Ural Federal University, Yekaterinburg, Russia, in 2011, and the M.Sc. degree from the Technical University of Munich in 2013, with masters thesis completed at NTT DOCOMOs European research laboratories (DOCOMO Euro-Labs) on mobility management in cellular networks. He is currently pursuing the Ph.D. degree with the Technical University of Munich. He has been with the Chair of Communication Networks, Department of Electrical and Computer Engineering, Technical University of Munich, since 2013. His research interests include multiple access protocols and machine-to-machine communications.

Jochen Guck received the Dipl.-Ing. degree in Ingenieurinformatik from the University of Applied Sciences Wuerzburg-Schweinfurt, Schweinfurt, Germany, in 2009, and the M.Sc. degree in electrical engineering from the Technical University of Munich, Munich, Germany, in 2011. In September 2012, he joined the Chair of Communication Networks at the Technical University of Munich as a member of the research and teaching staff. His research interests include real-time communication, industrial communication, software-defined networking, and routing algorithms.

Wolfgang Kellerer (M'96-SM'11) is a Full Professor with the Technical University of Munich (TUM), heading the Chair of Communication Networks at the Department of Electrical and Computer Engineering. Before, he was for over ten years with NTT DOCOMO's European Research Laboratories. He received his Dr.-Ing. degree (Ph.D.) and his Dipl.-Ing. degree (Master) from TUM, in 1995 and 2002, respectively. His research resulted in over 200 publications and 35 granted patents. He currently serves as an associate editor for IEEE Transactions on Network and Service Management and on the Editorial Board of the IEEE Communications Surveys and Tutorials. He is a member of ACM and the VDE ITG.