Paper published at the 24th IEEE European Test Symposium (ETS) in May 2019

Dependable Wireless Industrial IoT Networks: Recent Advances and Open Challenges

Fotis Foukalas*, Paul Pop*, Fabrice Theoleyre†, Carlo Alberto Boano‡, and Chiara Buratti§

*Dept. of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark – {fotisf,paupo}@dtu.dk
†CNRS, ICube, University of Strasbourg, France – theoleyre@unistra.fr
‡Institute of Technical Informatics, Graz University of Technology, Austria – cboano@tugraz.at
§Department of Electrical, Electronic, and Information Engineering, University of Bologna, Italy – c.buratti@unibo.it

***Abstract*—Industrial Internet of Things (IIoT) networks are considered the large-scale deployment of IoT devices for industrial applications such as smart manufacturing, harvesting and supply chain management. The Internet of Things (IoT) devices are typically connected over a wireless medium, given the large geographical distribution area and the increasing demand for flexible installations. In some cases, a combination of wired and wireless connectivity can be assumed as common practice. In both scenarios, wireless communications for IIoT networks is a fundamental component of the system architecture that needs to satisfy stringent requirements such as reliable connectivity and minimal delays. Therefore, the dependability of wireless communications for IIoT networks should be carefully studied to provide new solutions, which can guarantee that applications can meet their real-time and reliability requirements while optimizing the control capability of the overall network. This paper focuses on the dependable wireless communications in the IIoT networks, where wireless control and monitoring tasks need to meet stringent real-time and reliability constraints. After reviewing recent solutions and discussing their suitability for IIoT networks, we highlight the yet open challenges that needs to be tackled by both academia and industry.**

***Index Terms*—Industrial Internet of Things, Wireless Communication, Dependability.**

I. Introduction

Recent advances in wireless communication have increased the pervasiveness of wireless devices and enabled a plethora of new application domains. The use of wireless technologies is also expected to revolutionize industrial applications and to enable the creation of an Industrial Internet of Things (IIoT). Wireless IIoT solutions typically target industrial automation applications, such as the monitoring and control of various devices in a factory. Taking into account the industry 4.0 vision about deploying robotics for smart digital factories, wireless IIoT networks are becoming more interesting from a research and innovation point of view. On the one hand, with the growing number of wireless technologies and protocols for IIoT applications, it is becoming increasingly more important to benchmark their performance in order to understand which solutions are suitable for a given IIoT scenario and to allow meeting stringent application requirements. On the other hand, providing *dependability* is intrinsically challenging when deploying wireless IIoT networks.

In systems engineering, dependability is the ability to provide services that can defensibly be trusted within a given time-period [1]. Examples of dependability attributes are availability, reliability, safety, integrity and maintainability. In the next sections, we review several articles that are related to wireless IIoT networks and that address the aforementioned dependability attributes, highlighting recent trends and open challenges (Sect. II). We then discuss recent efforts in benchmarking IIoT systems and comparing their dependability (Sect. III). We further compare decentralized IIoT architectures with centralized ones (Sect. IV). Finally, we introduce recent work on long-range wireless industrial networks (Sect. V).

II. Dependable Wireless IIoT Networks: Overview and Open Challenges

A. Dependability overview in wireless IIoT networks

Industrial automation of smart digital factories represents the most challenging objective for future IIoT networking use cases. It mainly includes monitoring and control. In case of wireless IIoT networks, the design requirements include real-time monitoring and wireless industrial control [2]. Due to wireless fluctuations, both requirements are challenging to meet; however, solutions have been already provided. A common wireless IIoT network for monitoring and control is depicted in Fig.1, where many sensor devices transmit measurements to the wireless domain controller. This is known as convergecast scheduling and it is used by WirelessHART extensively [3]. In such a wireless IIoT network deployment, the latency-optimal convergecast scheduling problem is important to solve, where each sensor device has one packet to transmit to the controller, and the objective is to collect the data from all sensor devices at the controller in the minimum time. Further, the channel-constrained latency-optimal convergecast scheduling problem is also important to study, in which the number of channels available for convergecast is limited.

We now discuss about the availability, reliability, safety, integrity and maintainability attributes in IIoT-related studies:

- *Availability*: a dependability example is provided through a "schedulability" analysis for WirelessHART networks [4]. A key insight underlying analysis for mapping of the real-time transmission scheduling in WirelessHART networks to real-time multiprocessor scheduling is provided. The proposed analysis calculates a safe and tight upper bound of the end-to-end delay of every

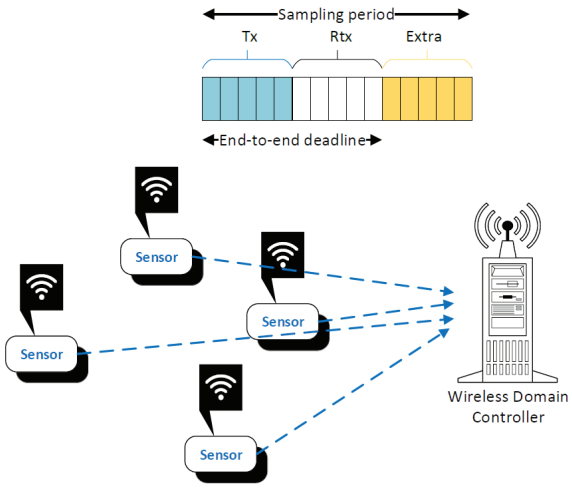

Fig. 1: Many-to-one communication in wireless IIoT networks.

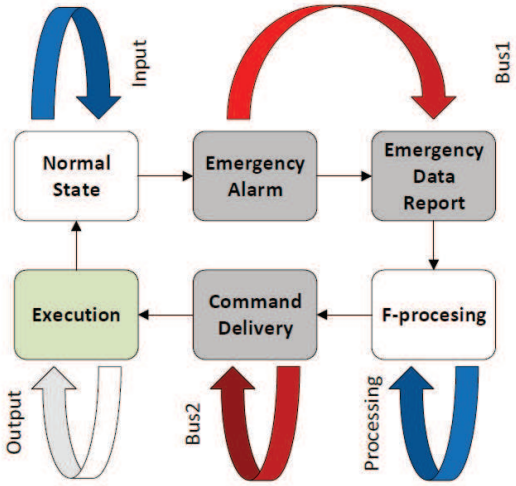

Fig. 2: Functional safety and response time modeling in wireless IIoT networks.

real-time periodic data flow in pseudo polynomial time. A key insight in this work is that we can map the multichannel fixed priority transmission-scheduling problem for WirelessHART networks to the fixed priority real-time CPU scheduling on a global multiprocessor platform.

- *Reliability*: when using Industrial Wireless Sensor Networks (IWSNs) in industrial applications such as real-time monitoring, it is important to consider both the sampling period and the end-to-end deadline [5]. The sampling period determines how often the monitoring application measures and transmits a new sample. The end-to-end deadline is measured from the time a packet is generated to the time it reaches its final destination.
- *Safety*: Yang et al. [6] present a novel framework for the support of safety-critical applications over mesh topology and multiple hops. The framework changes the time-triggered safety functions in IEC 61784-3-3 to event-triggered, which are more efficient for IWSNs (Fig. 2). To provide high priority for emergency safety events, an alarm-based method is also proposed for IWSNs. The most important metric for safety-critical applications is the safety function response time (SFRT).
- *Integrity*: in IIoT networks, integrity is often related to fault-tolerance in WSNs [7]. Fault detection in WSNs is a technique that identifies a fault when it occurs and pinpoints the type of fault and its location. Fault detection techniques can be classified into centralized, distributed, and hybrid. In centralized approaches, many algorithms are based on machine learning techniques.
- *Maintainability*: this attribute typically deals with the fault tolerance provided by cloud and fog computing architectures in IIoT services [8]. For example, in [9], the authors discuss an IoT platform that provides application fault tolerance through connection to multiple operating hubs. Devices can automatically connect to such operating hubs reporting similar events.

B. Open challenges

Several open challenges in terms of the aforementioned dependability attributes can be identified within a common core issue in several wireless IIoT use cases: the co-design of control, networking and computing in such a distributed networking environment [10]. IIoT networks are devoted to seamlessly interconnect complex industrial systems and to improve the reliability, efficiency, and productivity of industrial systems. The control, networking, and computing have already been identified as three key systems in IIoT networks. The integrated design of control, networking, and computing is hence an important and promising direction for future research [10]. Such a co-design is also mentioned as coupling between wireless communications: control therefore motivates a cyber-physical co-design approach that integrates wireless networks, and control designs. The coupling between real-time communication and control requires a cyber-physical co-design approach for a holistic optimization of control performance [2]. To this direction, authors in [8] proposed the implementation architecture of the SPSRP scheme, which first decoupled the computing control layer from the computing layer, and provided a programmable interface for IIoT operators. The computing control layer was decoupled from the computing layer and a programmable resources partitioning interface was provided. This novel architecture facilitated the IIoT to be more scalable to embrace industrial situation awareness. In addition to the co-design, we could distinguish the following open challenges:

- A timely analysis and processing of the massive data is also important to the time sensitive industrial systems.
- Visualization, virtualization and interoperability are key factors to enabling efficient management, effective utilization and timely maintenance.
- The mobility patterns and traffic patterns shall be considered in the design of wireless networks for industrial systems.

- An effective design and efficient deployment schemes, the mobility and network traffic patterns of machine devices in IIoT shall be understood, and all possible mobility scenarios should be covered.
- Traffic scheduling schemes can manage a large amount of traffic with the consideration of system performance metrics and requirements, including latency, utilization of network resources.
- Real-time information delivery that each traffic flow shall be delivered with latency guarantee requirements.

III. Benchmarking IIoT Systems and Protocols

Over the last decade, an increasing number of low-power wireless technologies and communication protocols have been developed in order to satisfy the requirements of a wide range of IoT applications. These technologies are largely different in nature: devices making use of IEEE 802.15.4, Bluetooth Low Energy, and ANT+ radios, for example, enable short-range multi-hop communications, whilst devices embedding LoRa, NB-IoT, Sigfox, and Weightless transceivers can be used to build long-range wide area networks. All these technologies specify different signal management functions, modulation schemes, data rates, channel bandwidths and separations. This results in largely-diverse performance and makes it hard to quantify or compare (i) the suitability for a given IIoT application, as well as (ii) the achievable dependability in terms of reliability, timeliness, and availability (energy-efficiency). Indeed, several communication protocols have been proposed in conjunction with these technologies to sustain a dependable performance, e.g., in noisy RF environments. Such protocols range from *multi-hop routing* solutions making use of time-slotted channel hopping [11], [12], [13], to solutions based on *synchronous transmissions* and *constructive interference* [14], [15], [16]. With so many options available, choosing the best constellation for an IIoT application with stringent dependability requirements in terms of reliability, availability, and timeliness can be very complex. This is especially true, as the parametrization of a given protocol for a specific technology strongly affects the achievable performance. Moreover, the set of configurable parameters to be chosen can be quite large, which exacerbates the problem even further.

A. Lack of Benchmarks for IIoT Protocols and Systems

Traditionally, benchmarking suites help system designers and industry practitioners to select the best combination of communication technology, network stack, and protocol parameters, to meet the desired application requirements. However, to date, there is a severe lack of benchmarks that evaluate the performance of low-power wireless networking protocols [17]. Existing IIoT benchmarks, such as EEMBC's IoTMark-BLE [18] and Cisco's TPCx-IoT [19] focus only on the energy-efficiency (availability) of BLE edge devices and on the data aggregation as well as storage capabilities of IoT gateways, respectively. Other benchmarks evaluate distributed stream processing systems hosted on cloud data-centers [20] or specifically analyse the dependability (timeliness) of different IoT platforms as a function of the employed processor [21]. When it comes to the performance of (I)IoT communication protocols, there is a severe lack of benchmarking suites. Low-power wireless protocols and systems are often tested experimentally on a large-scale using public testbeds with tens of nodes such as FlockLab [22], Indriya [23], and FIT IoT-LAB [24]. The extent to which the results for one system hold in another setup, however, remains rather unclear [25].

The problem is threefold: on the one hand, experimentation with wireless communications is intrinsically hard to reproduce and repeat, due to the strong impact of the surrounding environmental conditions as well as of the employed experimental setup. On the other hand, there is a lack of a *common framework* describing the configuration used to experimentally test the communication performance of the wireless system at hand. Such a configuration includes the test scenario (e.g., parameters such as traffic pattern and load) as well as the test environment (e.g., the density of the network in the testbed employed for the experiments) [25]. Furthermore, there is no *well-defined methodology* specifying how to plan, execute, and report on the obtained experimental results. Such a methodology should specify which metrics should be computed, which (and how much) data should be collected, the minimum number of trials to be performed, as well as how results should be processed and synthesized [26]. As a result, the performance of low-power wireless networking protocols is hardly comparable, and little is known about the dependability of low-power wireless protocols used to build IIoT applications. Furthermore, there is also a lack of quantitative comparisons of protocol performance in the presence of *harsh environmental conditions*, which are typical of industrial environments and often cause repeated failures hard to troubleshoot [27]. For example, temperature variations are known to affect communication performance [28], [29], [30], whereas radio interference typically increases packet loss, end-to-end latency, and energy consumption [31] – affecting in turn key dependability attributes of IIoT systems such as reliability, timeliness, and availability.

B. Recent Efforts in Benchmarking IoT Systems and Protocols

To fill this gap, the community has started to join forces in order to enable a more repeatable and reproducible experimental validation of low-power wireless IoT networking systems. For example, the Association for Computing Machinery (ACM) has introduced a badging system to make sure that the published research is documented, consistent, exercisable, and reproducible [32]. Moreover, an increasing number of scientific venues explicitly solicits the open availability of *datasets* and *tools* [33] as complementary materials. These initiatives show an important trend emphasizing the need for objective and reproducible comparisons, which can serve as a reference for the evaluation of products from the IoT industry. The 6TiSCH Open Data Action (SODA) [34] is a research project that aims to provide a reference benchmark for 6TiSCH solutions in order to automate and facilitate the comparison

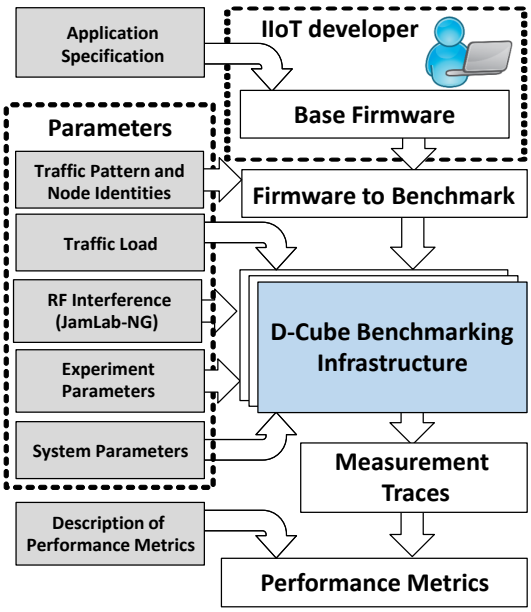

Fig. 3: Benchmarking an IIoT system using a testbed infrastructure based on D-Cube (adapted from [40]).

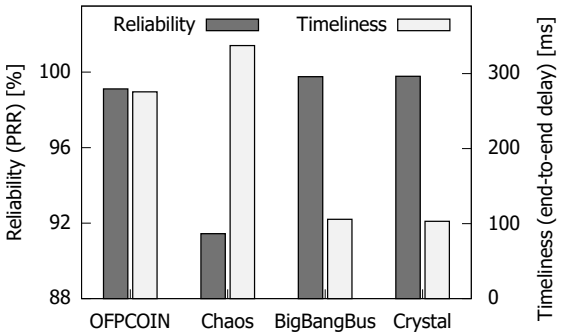

Fig. 4: Exemplary results obtained during the EWSN 2018 dependability competition, comparing the reliability and timeliness of different IoT protocols.

with future developments. SODA further aims to provide an open dataset with different IIoT-relevant scenarios.

Beyond these efforts, the *IoTBench initiative* [35], composed of several independent academic and industrial partners, explicitly aims to provide a set of tools and methods for benchmarking low-power IoT communication systems. Initial work within IoTBench has paved the way towards a common methodology for experimental evaluation [26] and a common framework describing the test configuration of wireless networking experiments [25]. Furthermore, parallel efforts within IoTBench have focused on developing (i) low-cost testbeds with benchmarking capabilities [36] and (ii) tools enabling the repeatable generation of harsh RF conditions [37]. These have been used – among others – to run public events to quantitatively benchmark the dependability of state-of-the-art wireless IIoT systems in harsh RF environments [38], and to stress-test the performance of Bluetooth Low Energy (BLE) in the presence of radio interference [39], as we describe next.

Low-cost testbeds with benchmarking capabilities. In order to allow the creation of low-cost infrastructures suitable to benchmark the performance of IIoT communication systems, Schuß et al. have developed *D-Cube* [36]. The latter is a low-cost tool that can be used to build or extend IoT testbeds such that one can accurately measure in hardware key dependability metrics such as end-to-end delay, reliability, and power consumption (availability), as well as to graphically visualize their evolution in real-time. D-Cube allows an automated, seamless, and fully user-customizable execution of low-power wireless networking evaluations, as shown in Fig. 3. Besides defining *input parameters* that directly characterize the configuration used to experimentally test the communication performance, e.g., (i) traffic parameters such as pattern and load; (ii) system parameters such as network density; (iii) experiment parameters such as duration and number of runs; D-Cube also allows a user to define *output metrics* quantifying the performance of the system under test. Such output metrics match the dependability attributes of interest, e.g., the IIoT system's reliability and timeliness, as well as availability (i.e., the overall energy consumption). Furthermore, D-Cube has been enriched with a *binary patching unit* that allows to split traffic pattern and node identities from the application specifications, which potentially allows an automated testing of different protocol parameters and configurations [40].

Tools to generate repeatable radio interference. The last years have seen a significant improvement in the performance of IIoT communication protocols and network stacks: to date, indeed, achieving 100% reliability in clean RF environments is possible at a relatively low price in terms of latency and energy [41], [42]. Achieving a high performance also in the presence of harsh RF conditions (that are typical of industrial environments), however, still remains a grand challenge. The main challenge here is to study and compare the performance of different IIoT communication protocols in the presence of RF interference. Indeed, there is a lack of tools enabling the controllable and repeatable generation of interference using, for example, Wi-Fi devices (the most prolific wireless technology operating in the 2.4 GHz ISM band). To fill this gap, Schuß et al. have developed *JamLab-NG* [37], a tool that enables the fine-grained control of individual link-layer transmissions of a Wi-Fi device. JamLab-NG avoids the uncontrollable delays introduced by the network stack, the operating system, and the clear channel assessment procedure. Furthermore, it allows to control radio settings such as transmission speed and packet length: these would traditionally be adapted by the radio firmware automatically – making it impossible to repeat the generated interference patterns [37].

Benchmarking the performance of IIoT solutions. D-Cube and JamLab-NG have been used together to run the EWSN (Embedded Wireless Systems and Networks) dependability competition series [38]. The latter is an international event organized to quantitatively benchmark the dependability of state-of-the-art academic and industrial IoT solutions for multi-hop data collection and multi-hop dissemination in the presence of

harsh Wi-Fi interference. Fig. 4 shows exemplary results from the 2018 edition for the reliability and timeliness of different IoT solutions in the presence of Wi-Fi interference resembling long bursts of fixed duration on a pre-defined channel. One can observe that solutions based on synchronous transmissions such as OFPCOIN [44], Chaos [45], BigBangBus [46], and Crystal [47] do achieve a high reliability with minimal end-to-end latency despite the congested RF environment. Further results can be found in [43]. In the 2019 edition, D-Cube was used to evaluate the performance of various IoT solutions using a different set of input parameters, such as the size of packets, the number of source and destination nodes, as well as the traffic load and period. Furthermore, JamLab-NG has been used to study the reliability of BLE in the presence of Wi-Fi interference [39], showing that an adaptation of the existing solutions is necessary to meet strict timeliness requirements – an important observation for BLE-based IIoT systems operating in congested RF environments.

IV. Decentralized vs. Centralized IIoT Architectures: Open Challenges

Industrial networks consist of a large set of sensors and actuators that exchange packets together, or via a set of border routers (Fig. 5). To provide strict guarantees, these networks rely on a strict transmission schedule to avoid collisions and to provide a high reliability. As wireless transmissions are unreliable by nature, most standards rely on slow frequency hopping techniques to improve the reliability [48]. As a result, one packet and its retransmissions do not use the same channel, reducing the probability to be affected repetitively by external interference.

Thus, to provide a set of guarantees, one needs to carefully build a 2D scheduling matrix [49]. The channel offsets (frequency) are represented vertically, while the timeslots are represented horizontally (Fig. 5). A cell in the matrix denotes, for a pair of devices, the timeslot and the channel offset used to transmit a packet. For safety-critical applications, the schedule has to be constructed such that it provides an upper-bound for the end-to-end delay. Typically, cells for retransmissions are contiguous in Fig. 5 to reduce the delay.

Because IIoT rely on wireless transmissions, the scheduling algorithm has to respect half-duplex constraints, a device cannot receive or transmit during the same timeslot through different channel offsets [50]. Similarly, the scheduler has to assign the same cell only two non-interfering transmitters.

Thus, we face to two possible approaches:

Centralized: a central entity (aka the Controller) has a complete view of the network's characteristics and requirements, and computes a global schedule. Then, the controller notifies individually each device with the part of the schedule it has to follow (as receiver or transmitter);

Distributed: when a pair of nodes wants to exchange packets, they need to define autonomously the timeslots and channel offsets to use. They have to avoid creating collisions with already scheduled transmissions in the vicinity.

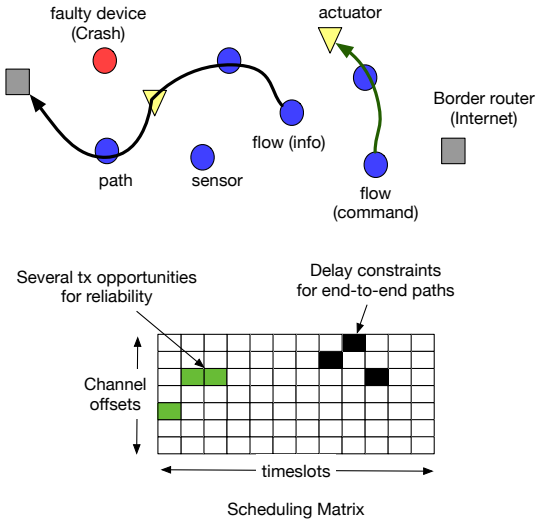

Fig. 5: Network topology and scheduling matrix for an industrial network.

A. Open Problems for Centralized Scheduling

Centralized scheduling algorithms can construct a schedule able to respect a set of guarantees. For instance, we may provision additional bandwidth for retransmissions to respect end-to-end reliability constraints [51]. Tools exist to verify that a schedule can respect weakly hard real-time constraints [52].

1) Monitoring: Constructing an efficient schedule requires to have a precise view of the network properties. For instance, the scheduler needs:

reliability: each link has to monitor the Packet Error Rate so that end-to-end reliability can be esimated;

traffic: the controller needs to know when packets are generated, and which is the destination;

network topology: to select the right set of forwarding devices in multihop topologies, all links have to be reported.

Practically, this represents a huge amount of control traffic, which has to be delivered reliably to the controller. To reduce the load, control information can be piggybacked in the data traffic, to increase the packet length instead of the number of packets [53]. To the best of our knowledge, no research paper describes experimental results reporting the efficiency of a centralized schedule executed on top of a large scale multihop topology.

2) Network Formation and Convergence: To bootstrap, the devices which join the network need to have specific transmission opportunities to join the controller. In the Software Defined Network architecture, a specific control plane exists, forwarding all the packets to the controller that configures later the network to forward the data packets [55]. Typically, best-effort cells are reserved for the network formation. However, a large number of devices implies also a large convergence delay since the Slotted-Aloha method leads to many collisions [56]. Reliability is only achieved after the network has converged, i.e., each data flow has dedicated resources.

3) Integrity (fault-tolerance): Notifying the controller that needs to recompute a novel schedule requires a significant amount of time. Thus, fault-tolerance has to be considered directly when constructing the schedule. Graph routing techniques creates a collection of paths toward the border routers [57]. By allocating the same resource through different paths, the network can handle a node's failure without affecting reliability. Anycast scheduling consists in exploiting the broadcast nature of wireless transmissions to schedule several receivers for a transmission to handle a fault [54].

After having detected a fault, the controller needs a long time (e.g., a few minutes) to update its network level view, to recompute a novel schedule, and to push the updates reliably to all the devices. Thus, faults have to be sufficiently interspaced to let the network re-converge.

B. Open Problems for Distributed Scheduling

For a distributed strategy, we need two complementary mechanisms:

a routing protocol so that each device knows a next hop toward the destination (often, the border router). RPL [58] represents the most common standard routing protocol for low-power and lossy networks;

scheduling techniques to decide which cells to use for any pair of transmitter/receiver. Typically, a protocol such as 6P defines the way to negotiate the cells to use [59].

1) Reliability (collision Avoidance): Because each pair of device has only a partial knowledge of the cells previously allocated, distributed scheduling may lead to collisions. Several existing mechanisms try to identify collisions by identifying the cells that exhibit an abnormally low Packet Delivery Ratio [60]. However, these solutions assume stable conditions (external interference, link reliability) with periodic traffic. In particular, bursty traffic makes the collisions less repetitive, i.e., only one part of the cells are busy, introducing a bias in the reliability estimation. Typically, detecting collisions for backup cells may be very challenging, although they are absolutely necessary to respect high reliability constraints.

2) Safety: Most critical applications require that a packet is delivered within a given deadline. Unfortunately, tackling such an end-to-end delay constraint is particularly challenging in multi-hop topologies, since fixing the instant of emission for the first hop may create a domino effect [61]. Only a few scheduling algorithms have been proposed to tackle this problem [62]. They consist in reserving consecutive slots in priority for the different devices along a path. In the same way, additional slots have to be reserved consecutively for retransmissions. However, such approach also increases the number of collisions, and complicates significantly the schedule's updates. Indeed, allocating another cell for retransmissions may require to re-schedule the rest of the path toward the destination.

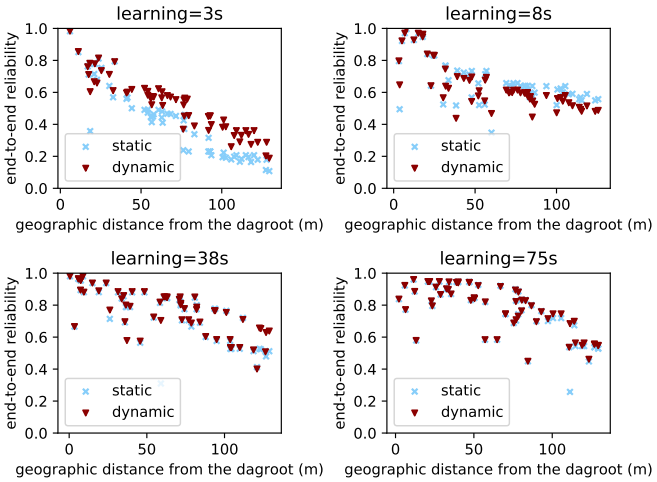

Fig. 6: End-to-end PDR when using the number of transmissions as routing metric with static (centralized) vs. dynamic (distributed) approaches.

C. Experimental Evaluation

We exploit here the dataset obtained in an indoor testbed, monitoring continuously 267 links during 90 minutes[1]. In particular, we split the whole experiment in different time windows of different size (i.e., the *learning* duration). We focus here on the reliability characteristic, measuring the end-to-end reliability when not considering retransmissions with static vs. dynamic routes (Fig. 6). For the static case, the routes are computed once, by minimizing the cumulative number of transmissions, when using the first time window to compute the routing cost. For the dynamic case, routes are updated at the end of each time window, with the statistics from the last time window. We can see that using time windows that are too short (e.g., 3 seconds) is inefficient: the statistics are not sufficiently *smoothed* and the end-to-end reliability is quite low. However, 30 seconds are sufficient to compute stable link quality metrics, and to identify reliable paths. Besides, the static and dynamic cases provide a very similar reliability in that case. Thus, exploiting static schedules seems reasonable, given that the link characteristics are sufficiently stable for so long durations.

D. Perspectives

Mixing distributed and centralized solutions seems to represent a promising solution. Recently, MABO-TSCH [63] has combined a centralized scheduling algorithm with a localized process to decide which channels to use for each transmission. The bundle concept of 6TiSCH [55] allows the controller to allocate blocks of cells to each device. Then, each device is in charge of managing its bundles, allocating transmission opportunities to its neighbors. Collisions can be avoided if the controller allocates the bundles to different areas. By properly distributing one part of the decisions, we may be able to

[1]https://github.com/ftheoleyre/fitiotlab-multichannel-dataset

allocate enough resources to enable safety. The network would also be more fault-tolerant to guarantee the integrity, by taking local decisions.

V. Dependable Wireless Solutions for the IIoT

In the past years, Zigbee over IEEE 802.15.4 [64] has dominated the IoT scene, with application even to (some) industrial contexts. The advantages of IEEE 802.15.4-based solutions over the Wi-Fi family of standards lay on better energy efficiency, which is of particular relevance when nodes are equipped with tiny batteries, as in the case of IIoT applications requiring nodes to be deployed over machines where cables have to be avoided. The IEEE 802.15.4, used over the 2.4 GHz ISM band, allows transmissions at a bit rate of 250 kbit/s, which can be useful for a number of industrial applications, especially when monitoring machines equipped with nodes that cannot be connected to the energy grid. However, this technology is based on distributed protocols, both at the medium access control (MAC) layer and at the network level, making its dependability arguable. It suffers from several limitations that degrade its performance: the unbounded delay, MAC unreliability and interference robustness [65]. To overcome these limitations, 6TiSCH has been proposed, and can provide a relevant solution for short-range, multihop topologies.

In the past few years, a number of solutions have emerged in the category of long range communication systems. We differentiate among those who have been standardized by 3GPP (like LTE-M and NB-IoT [66], [67]) and work over frequency bands used by telecom operators, and those that use ISM (license-exempt) bands, typically at 868 or 900 MHz, like LoRa [68] and Sigfox. LoRa is a proprietary solution that can be implemented by a network provider that ensures coverage through the LoRaWAN protocol stack and architecture, or through a dedicated ad-hoc protocol suite designed to fulfill specific requirements. LoRa defines a transmission technique which is very flexible: through the configuration of the Spreading Factor (SF) parameter, nodes can trade transmission ranges with bit rate. The smaller is the SF, the higher is the bit rate while the range decreases, increasing reliability. However, due to the limited bandwidth available in the 868 or 900 MHz ISM bands, the bit rate is never larger than a few tens of kbit/s. Very recently, the company that holds LoRa's patents has delivered a solution running at 2.4 GHz, which is characterized by the same degree of flexibility, while using larger bandwidths (hence providing higher bit rates), resulting in lower end-to-end delays. In other words, LoRa at 2.4 GHz can provide the same performance offered by Zigbee over IEEE 802.15.4, with the possibility of using the SF as a parameter to trade dynamically range with data rates. ¿¿¿¿¿¿¿ ca2486a5a76a98cf73dd0acfdd69d758bc71b4e6

A. LoRaIN: A LoRa-Based Dependable Solution for IIoT

Since January 2018, when the SX1280 chip implementing LoRa at 2.4 GHz was delivered by Semtech, a number of institutions are prototyping multi-hop protocols to be used over such transmission technique. Among them, the University of Bologna, under commitment by a local industry, is designing a reliable, flexible and robust protocol stack for dependable IIoT. In particular, the scope is to conceive a system where sensor nodes (up to 1000) are deployed over a machine, with a density of up to 50 nodes/sqm, where part or all of them apply harvesting techniques to accumulate the energy needed to communicate the sensor measurements and receive commands or strings of bytes to be recorded in the local memory.

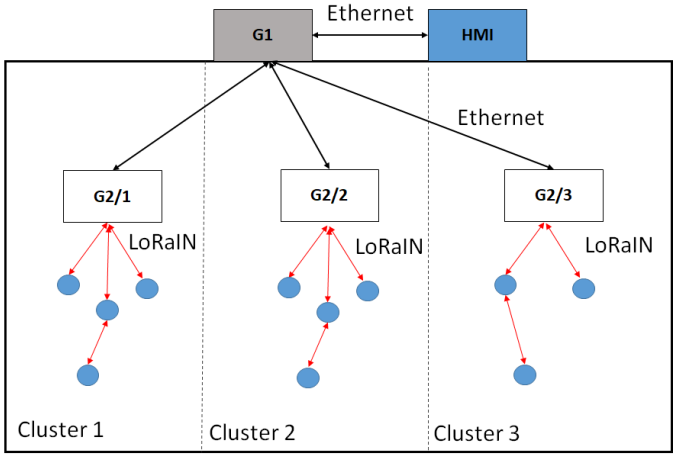

Fig. 7: The LoRaIN Architecture.

The proprietary protocol (denoted as LoRaIN: LoRa Industrial Network) is based on a hierarchical network architecture (see Fig. 7), where a Level 1 Gateway (G1), linked to the HMI (Human Machine Interface) of the industrial machine, manages the commands coming from it, and implements the algorithms used to control the network of sensor nodes (denoted as tags). The G1 node is connected through Ethernet to few Level 2 Gateways (G2s), each of them being responsible for the communication with the tags in a different area, denoted as cluster. Communication through tags and G2 happen via the LoRaIN protocol. Every G2 uses two frequency bands: one is common to all G2s and it is used for tag association purposes, while the other is different from cluster to cluster, to ensure that inter-cluster interference is avoided. Within a cluster, up to 50 tags communicate according to a schedule defined by G1 and managed by G2s.

A centralized approach is used to manage the access to the channel: G2s periodically send beacon packets to maintain synchronization and to assign time slots to nodes as soon as a request of communication is started at the HMI and forwarded to G2s via G1. Each communication frame (where a frame is the time interval between two subsequent beacons) is split into sub-intervals: one is left for the tag to harvest the energy needed for the following steps, one for the uplink and/or downlink communication. Every beacon describes the structure of the following frame, which is dynamically defined according to the commands received from the HMI, and the current network state. The duration of frames (inter-beacon

time), and of the two sub-intervals (for energy harvesting and for communication) are dynamically computed (at G1) depending on the commands received, the amount of bytes to be sent/received.

Through the time slots scheduling, intra-cluster interference is completely absent, as for the inter-cluster interference is avoided through the implementation of a frequency reuse strategy. The latter approach ensures that the network of tags is highly reliable. Moreover, LoRaIN takes advantage of the degree of flexibility offered by LoRa, by chosing SF according to the path losses measured from the tags to the relevant G2. The more the cluster G2 is deployed in a proper position, the better are the links and higher the bit rates, with consequent smaller delays in the provision of the monitored data.

As far as delays are concerned, the ability of LoRaIN to provide data from the tags with stringent delay requirements (compatible with a bit rate no larger than approximately 250 Kbit/s), depends on the interplay between the type of energy harvesting technique implemented and the amount of data to read/write. For those tags that are battery-driven, the first sub-interval of each frame can be eliminated and the HMI can receive data from a selected tag within about 200 ms, a delay compatible to many monitoring applications. Delays are larger when the miniaturization of tags does not allow the use of a (even tiny) battery. The main issue that is subject to some level of unpredictability, thus representing a threat to the level of reliability/dependability, is external interference.

B. LoRaIN: Testing Reliability

In this subsection, the reliability of the network is tested in terms of Packet Error Rate (PER), being the percentage of packets lost in the LoRaIN network (working at 2.4 GHz), when affected by the interference generated by an IEEE 802.11g network.

For experiments we have used the TP-Link TL-WA830RE as access point, configured to work in IEEE 802.11g mode with maximum available transmit power and a maximum rate allowed of 54 Mbit/s. To generate a constant Wi-Fi occupancy we used two laptops working as video streaming client and server (VLC Media player was running at both laptops). An HTTP (HyperText Transfer Protocol) server was set up to broadcast an mp4 video in on-demand fashion. The experimental setup is shown in Fig. 8: Server and AP were connected exploiting an Ethernet cable, while the client was communicating to the AP via the wireless connection through IEEE 802.11g. As for LoRa, we used the SX1280 [69] devices; the transmitter-receiver distance was set to 3.2 m and both devices were at an height of 0.5 m from the floor.

In order to characterize the interference level suffered by the LoRa receiver, we estimated the Signal-to-Interference Ratio (C/I) as follows. A Wi-Spy device was located near the LoRa receiver to measure the interfering power, I, while the useful received power, C, was directly measured by the LoRa receiver using the received signal strength indicator (RSSI). Finally, we used the Chanalyzer to measure the channel utilization, defined as the percentage of time the received signal was

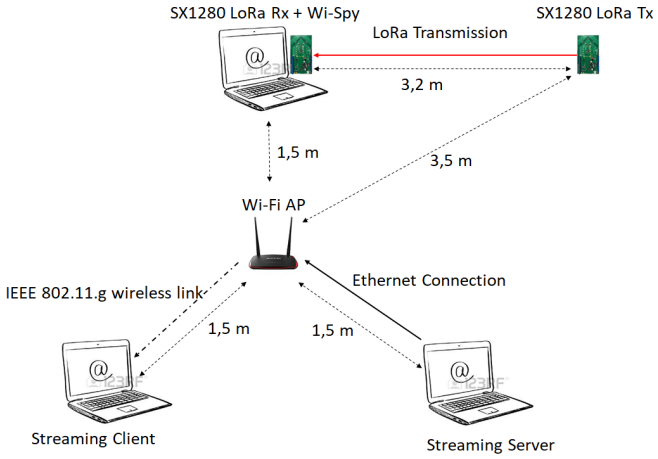

Fig. 8: Interference Evaluation: Experimental setup.

above a threshold of -105 dBm in the considered channel for a given interval of time (set equal to the duration of a single experiment). The PER has been computed considering the transmission of 1000 packets.

In Fig. 9 we show the measured PER as a function of LoRa's SF for different values of C/I. The channel occupancy was 20%. The carrier frequency of LoRa signal was set to 2,452 GHz, that is the central frequency of channel 9 used by the IEEE 802.11g network (completely overlapped channels). The bandwidth selected for LoRa transmission was set to 203 kHz. As expected, the PER presents a maximum value due to the following. On the one hand, by increasing the spreading factor, the time on air of LoRa increases, bringing to larger collision probability; on the other hand, large values of spreading factor result in a modulation which is more robust to interference. In fact, when setting SF=12 the PER becomes zero, except for the case $C/I = -38$ dB, where the SIR is too low and the PER cannot reach zero. In conclusion, by properly setting the spreading factor, LoRaIN demonstrated to have high reliability; however a trade-off between robustness to interference and end-to-end delay should be found, since the increasing of SF results in longer time-on-air, and therefore delays. However, in all cases delays are bounded.

VI. Conclusions

This paper provides an overview on the recent advances and open challenges in building dependable wireless IIoT networks. We first describe efforts in benchmarking wireless IIoT networking technologies and protocols. Particular studies have been discussed and the dependability aspects of such technologies have been highlighted. We then present a study on decentralized and centralized IIoT architectures and an overview of recent efforts in the area of long-range wireless technologies that are suitable for IIoT settings. As we point out throughout the paper, there are still several open challenges that need to be addressed in order to deploy dependable wireless systems in future industrial automation applications:

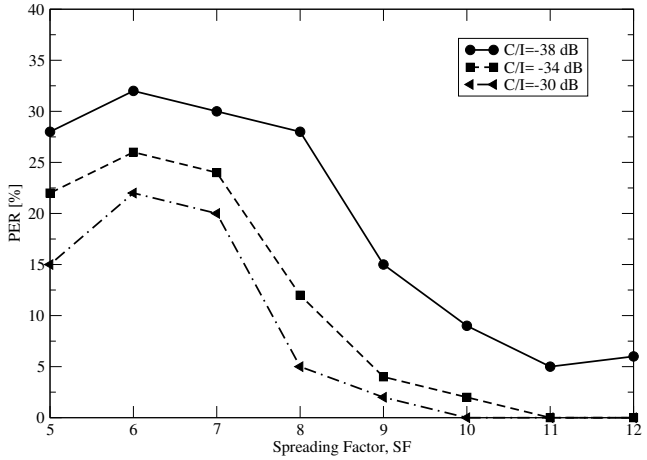

Fig. 9: Packet Error Rate as a function of the SF in a LoRa network affected by the interference generated by an IEEE 802.11g network, for different SIR values.

only by solving those one can exploit the flexibility offered by wireless technology and build a reliable and safe IIoT.

REFERENCES

[1] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing, 1(1), 2004.

[2] C. Lu, et al., Real-Time Wireless Sensor-Actuator Networks for Industrial Cyber-Physical Systems, Proc. of the IEEE, vol. 104, no. 5, 2016.

[3] H. Zhang, P. Soldati and M. Johansson, Performance Bounds and Latency-Optimal Scheduling for Convergecast in WirelessHART Networks, IEEE Trans. Wirel. Commun. vol. 12, no. 6, 2013.

[4] A. Saifullah, Y. Xu, Ch. Lu and Y. Chen, End-to-End Communication Delay Analysis in Industrial Wireless Networks, IEEE Trans. Comp., vol. 64, no. 5, 2015.

[5] D. Yang, et al., Assignment of Segmented Slots Enabling Reliable Real-Time Transmission in Industrial Wireless Sensor Networks, IEEE Trans. Ind. Electr., vol. 62, no. 6, 2015.

[6] D. Yang, J. Ma, Y. Xu and M. Gidlund, Safe-WirelessHART: A Novel Framework Enabling Safety-Critical Applications Over Industrial WSNs, IEEE Trans. Ind. Informatics, vol. 14, no. 8, 2018.

[7] S. Hu and G. Li, Fault-tolerant clustering topology evolution mechanism of wireless sensor networks, IEEE Access, vol. 6, 2018.

[8] G. Li, J. Wu, J. Li, K. Wang, and T. Ye, Service Popularity-Based Smart Resources Partitioning for Fog Computing-Enabled Industrial Internet of Things, IEEE Trans. Ind. Informatics, vol. 14, no. 10, 2018.

[9] D. Terry, Toward a new approach to IoT fault tolerance, IEEE Computer Magazine, vol. 49, no. 8, 2016.

[10] X. Hu, W. Yu, D. Griffith, and N. Golmie, A Survey on Industrial Internet of Things: A Cyber-Physical Systems Perspective, IEEE Access Journal, vol. 6, 2018.

[11] J. Shi, M. Sha, and Z. Yang, DiGS: Distributed Graph Routing and Scheduling for Industrial Wireless Sensor-Actuator Networks, in Proc. of the 38th Conf. on Distributed Computing Systems (ICDCS), 2018.

[12] R. Tavakoli, M. Nabi, T. Basten, and K. Goossens, Dependable Interference-Aware Time-Slotted Channel Hopping for Wireless Sensor Networks, ACM Transactions on Sensor Networks, vol. 14, no. 1, 2018.

[13] I. Hosni and F. Theoleyre, Self-healing Distributed Scheduling for End-to-end Delay Optimization in Multihop Wireless Networks with 6TiSCH, Computer Communications, vol. 110, pp. 103–119, 2017.

[14] T. Istomin, M. Trobinger, A. L. Murphy, and G. P. Picco, Interference-resilient Ultra-low Power Aperiodic Data Collection, in Proc. of the 17th Conf. on Information Processing in Sensor Networks (IPSN), 2018.

[15] F. Mager et al., Feedback Control Goes Wireless: Guaranteed Stability over Lowpower Multi-hop Networks, CoRR, Tech. Rep. ArXiv:1804.08986, 2018.

[16] T. Chang, T. Watteyne, X. Vilajosana, and P. H. Gomes, Constructive Interference in 802.15.4: A Tutorial, IEEE Communications Surveys and Tutorials, 2018.

[17] S. Duquennoy et al., A Benchmark for Low-power Wireless Networking, in Proc. of the 14th ACM Conference on Embedded Networked Sensor Systems (SenSys), poster session, 2016.

[18] IoTMark-BLE: an EEMBC Benchmark, http://www.eembc.org/iotmark.

[19] The TPCx-IoT Benchmark for IoT Gateway Systems, http://www.tpc.org/tpcx-iot/.

[20] A. Shukla, S. Chaturvedi, and Y. Simmhan, RIoTBench: A Real-time IoT Benchmark for Distributed Stream Processing Platforms, CoRR, Tech. Rep. ArXiv:1701.08530v1, 2017.

[21] C. P. Kruger and G. P. Hancke, Benchmarking Internet of Things Devices, in Proc. of the 12th IEEE Conf. on Industrial Informatics (INDIN), 2014.

[22] R. Lim, F. Ferrari, M. Zimmerling, C. Walser, P. Sommer, and J. Beutel, FlockLab: A testbed for distributed, synchronized tracing and profiling of wireless embedded systems, in Proc. of the 12th Conference on Information Processing in Sensor Networks (IPSN), 2013.

[23] M. Doddavenkatappa, M. C. Chan, and A. Ananda, Indriya: A lowcost, 3D wireless sensor network testbed, in Proc. of the 7th Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom), 2011.

[24] C. Adjih et al., FIT IoT-LAB: A Large Scale Open Experimental IoT Testbed, in Proc. of the 2nd World Forum on the Internet of Things (WF-IoT), 2015.

[25] C. A. Boano et al., Towards a Benchmark for Low-power Wireless Networking, in Proc. of the 1st Workshop on Benchmarking Cyber-Physical Networks and Systems (CPSBench), 2018.

[26] R. Jacob et al., Towards a Methodology for Experimental Evaluation in Low-Power Wireless Networking, in Proc. of the 2nd Workshop on Benchmarking Cyber-Physical Systems and Internet of Things, 2019.

[27] U. Wetzker et al., Troubleshooting Wireless Coexistence Problems in the Industrial Internet of Things, in Proc. of the 14th IEEE/IFIP Conference on Embedded and Ubiquitous Computing (EUC), 2016.

[28] K. Bannister, G. Giorgetti, and S. K. Gupta, Wireless Sensor Networking for Hot Applications: Effects of Temperature on Signal Strength, Data Collection and Localization, in Proc. of the 5th Workshop on Embedded Networked Sensors (HotEmNets), 2008.

[29] C. A. Boano, K. Römer, and N. Tsiftes, Mitigating the Adverse Effects of Temperature on Low-Power Wireless Protocols, in Proc. of the 11th Conference on Mobile Ad hoc and Sensor Systems (MASS), 2014.

[30] C. A. Boano, M. Cattani, and K. Römer, Impact of Temperature Variations on the Reliability of LoRa: An Experimental Evaluation, in Proc. of the 7th Conference on Sensor Networks (SENSORNETS), 2018.

[31] C. A. Boano and K. Römer, External radio interference, in Radio Link Quality Estimation in Low-Power Wireless Networks, SpringerBriefs in Electrical and Computer Engineering Cooperating Objects, 2013.

[32] Association for Computing Machinery, Artifact Review and Badging, https://www.acm.org/publications/policies/artifact-reviewbadging.

[33] Data: Acquisition To Analysis, Workshop in conjunction with ACM SenSys, https://workshopdata.github.io/DATA2018.

[34] M. Vučinić, M. Pejanović-Djurišić, and T. Watteyne, SODA: 6TiSCH Open Data Action, in Proc. of the 1st Workshop on Benchmarking Cyber-Physical Networks and Systems (CPSBench), 2018.

[35] IoTBench: A Community Effort to Better Evaluate and Compare Low-Power Wireless Networks, https://www.iotbench.ethz.ch/.

[36] M. Schuß et al., A Competition to Push the Dependability of Low-Power Wireless Protocols to the Edge, in Proc. of the 14th Conf. on Embedded Wireless Systems and Networks (EWSN), 2017.

[37] M. Schuß, C. A. Boano, M. Weber, M. Schulz, M. Hollick, and K. Römer, JamLab-NG: Benchmarking Low-Power Wireless Protocols under Controlable and Repeatable Wi-Fi Interference, in Proc. of the 16th Conf. on Embedded Wireless Systems and Networks (EWSN), 2019.

[38] C. A. Boano, et al., EWSN Dependability Competition: Experiences and Lessons Learned, IEEE Internet of Things Newsletter, 2017.

[39] M. Spörk, C. A. Boano, and K. Römer, Improving the Timeliness of Bluetooth Low Energy in Noisy RF Environments, in Proc. of the 16th Conf. on Embedded Wireless Systems and Networks (EWSN), 2019.

[40] M. Schuß, C. A. Boano, and K. Römer, Moving Beyond Competitions: Extending D-Cube to Seamlessly Benchmark Low-Power Wireless Systems, in Proc. of the 1st Workshop on Benchmarking Cyber- Physical Networks and Systems (CPSBench), 2018.

[41] S. Duquennoy, J. Eriksson, and T. Voigt, Five-Nines Reliable Downward Routing in RPL, CoRR, Tech. Rep. ArXiv:1710.02324v1, 2017.

[42] C. Herrmann et al., Mixer: Efficient Many-to-All Broadcast in Dynamic Wireless Mesh Networks, in Proc. of the 16th ACM Conference on Embedded Networked Sensor Systems (SenSys), 2018.
[43] EWSN 2018 Dependability Competition – Official blog, https://iti-testbed.tugraz.at/blog/tag/ewsn2018/.
[44] X. Ma et al., Competition: Using Enhanced OFPCOIN to Monitor Multiple Concurrent Events under Adverse Conditions, in Proc. of the 15th Conf. on Embedded Wireless Systems and Networks (EWSN), 2018.
[45] B. Al Nahas and O. Landsiedel, Competition: Aggressive Synchronous Transmissions with In-network Processing for Dependable All-to-All Communication, in Proc. of the 15th Conf. on Embedded Wireless Systems and Networks (EWSN), 2018.
[46] A. Escobar et al., Competition: BigBangBus, in Proc. of the 15th Conf. on Embedded Wireless Systems and Networks (EWSN), 2018.
[47] M. Trobinger et al, Competition: CRYSTAL Clear: Making Interference Transparent, in Proc. of the 15th Conf. on Embedded Wireless Systems and Networks (EWSN), 2018.
[48] T. Watteyne, A. Mehta, and K. Pister, Reliability through frequency diversity: Why channel hopping makes sense, in Proceedings of the 6th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN), 2009.
[49] R. Teles Hermeto, A. Gallais, and F. Theoleyre, Scheduling for IEEE802.15.4-TSCH and Slow Channel Hopping MAC in Low Power Industrial Wireless Networks, Comput. Commun., vol. 114, no. C, 2017.
[50] 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 IEEE 802.15.4e TSCH, IEEE Sensors Journal, vol. 13, no. 10, 2013.
[51] F. Dobslaw, T. Zhang, and M. Gidlund, End-to-end reliability-aware scheduling for wireless sensor networks, IEEE Transactions on Industrial Informatics, vol. 12, no. 2, 2016.
[52] J. B. Schmitt and U. Roedig, Sensor network calculus – a framework for worst case analysis, in Distributed Computing in Sensor Systems, Springer Berlin Heidelberg, 2005.
[53] G. Gaillard et al., Monitoring kpis in synchronized ftdma multi-hop wireless networks, in Proc. of the Wireless Days (WD), 2016.
[54] I. Hosni, and F. Theoleyre, "Adaptive k-cast Scheduling for high-reliability and low-latency in IEEE802.15.4-TSCH", Conference on Ad-hoc, Mobile, and Wireless Networks (ADHOC-NOW), 2018.
[55] P. Thubert, M. R. Palattella, and T. Engel, 6TiSCH centralized scheduling: When SDN meet IoT, in Conference on Standards for Communications and Networking (CSCN), 2015.
[56] R. Teles Hermeto, A. Gallais, and F. Theoleyre, On the (over)-reactions and the stability of a 6TiSCH network in an indoor environment, in Proc. of the Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, 2018.
[57] J. Song, et al., WirelessHART: Applying wireless technology in real-time industrial process control, in Real-Time and Embedded Technology and Applications Symposium, 2008.
[58] P. Thubert et al., RPL: IPv6 routing protocol for low-power and lossy networks, IETF, RFC 6550, 2012.
[59] Q. Wang, et al., 6TiSCH operation sublayer (6top) protocol (6p), IETF, RFC 8480, 2018.
[60] K. Muraoka, T. Watteyne, N. Accettura, X. Vilajosana, and K. S. J. Pister, Simple distributed scheduling with collision detection in TSCH networks, IEEE Sensors Journal, vol. 16, no. 15, 2016.
[61] G. Gaillard, D. Barthel, F. Theoleyre, and F. Valois, Kausa: Kpi-aware scheduling algorithm for multi-flow in multi-hop iot networks, in Adhoc, Mobile, and Wireless Networks, Springer Publishing, 2016.
[62] I. Hosni and F. Theoleyre, Self-healing distributed scheduling for end-to-end delay optimization in multihop wireless networks with 6TiSCH, Computer Communications, vol. 110, 2017.
[63] P. H. Gomes et al., Mabo-tsch: Multihop and blacklist-based optimized time synchronized channel hopping, Transactions on Emerging Telecommunications Technologies, vol. 29, no. 7, 2018.
[64] IEEE, Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), in IEEE Std 802.15.4-2011, 2011.
[65] A. Nikoukar, et al., Low-Power Wireless for the Internet of Things: Standards and Applications, IEEE Access, 2018.
[66] R. Ratasuk, B. Vejlgaard, N. Mangalvedhe and A. Ghosh, NB-IoT system for M2M communication, in Prof. of the Wireless Communications and Networking Conference Workshops (WCNCW), Doha, 2016.
[67] L. Feltrin, G. Tsoukaneri, M. Condoluci, C. Buratti, T. Mahmoodi, M. Dohler, R. Verdone, Narrowband IoT: A Survey on Downlink and Uplink Perspectives, in IEEE Wireless Communications, vol. 26, no. 1, 2019.
[68] L. Feltrin, C. Buratti, E. Vinciarelli, R. De Bonis and R. Verdone, LoRaWAN: Evaluation of Link- and System-Level Performance, in IEEE Internet of Things Journal, vol. 5, no. 3, 2018.
[69] Semtech SX1280 datasheet, https://www.semtech.com/products/wireless-rf/24-ghz-transceivers/sx1280.