The Impact of Networking Protocols on Massive M2M Communication in the Industrial IoT

Cenk Gündoğan, Peter Kietzmann, Martine S. Lenders, Hauke Petersen, Michael Frey,
Thomas C. Schmidt, Felix Shzu-Juraschek, and Matthias Wählisch

***Abstract*—Common use cases in the Industrial Internet of Things (IIoT) deploy massive amounts of sensors and actuators that communicate with each other or to a remote cloud. While they form too large and too volatile networks to run on ultra-reliable, time-synchronized low-latency channels, participants still require reliability and latency guaranties. We elaborate this for safety-critical use cases. This paper focuses on the effects of networking protocols for industrial communication services. It analyzes and compares the traditional Message Queuing Telemetry Transport for Sensor Networks (MQTT-SN) with the Constrained Application Protocol (CoAP) as a current IETF recommendation, and also with emerging Information-centric Networking (ICN) approaches, which are ready for deployment. Our findings indicate a rather diverse picture with a large dependence on deployment: Publish-subscribe protocols are more versatile, whereas ICN protocols are more robust in multi-hop environments. MQTT-SN competitively claims resources on congested links, while CoAP politely coexists on the price of its performance.**

***Index Terms*—Industrial Internet of Things, 5G, constrained environment, MQTT, CoAP, NDN, performance evaluation**

I. Introduction

The Internet of Things (IoT) is evolving by an increasing number of controllers in the field that is augmented with network interfaces which speak IP. Emerging industrial IoT (IIoT) deployments are often stimulated by adding online services to already existing systems for the sake of additional features and benefits. Such devices usually connect to power, use common broadband links, and adopt the old Message Queuing Telemetry Transport (MQTT) protocol [1] for publishing IoT data to a remote cloud. The prevalent use case forecast for the IoT, though, consists of billions of constrained sensors and actuators that are mainly not cabled to power, but mobile and connected via low power wireless links. The key target of the IoT will be data generated from massive amounts of tiny, cheap *things* that are severely challenging the current way of connecting to the Internet.

A number of approaches allow the creation of networks that can tackle these challenges, some of which are part of the recent 5G [2] efforts. 5G allows companies to create their own private networks on site. Companies can make use of a key 5G concept called network slicing. Network slicing enables the creation of sub-networks for specific services and users that can have specific 5G network parameters such as end-to-end latency, maximum throughput, and traffic density. It allows companies to deploy ultra-reliable low-latency networks for critical infrastructure by exploiting time-slotted wireless link technologies. It also supports massive machine type communication (mMTC) services to integrate billions of things using contention-based wireless access, which is in the focus of this article.

This new class of connected devices cannot be integrated into today's Internet infrastructure without technologies that bridge the scale. The IETF has designed a suite of protocols for successfully serving the needs of a constrained IoT. IPv6 adaptation layers such as 6LoWPAN [3] enable a deployment on constrained links (*e.g.,* IEEE 802.15.4 [4]), which the Routing Protocol for Low-Power and Lossy Networks (RPL) [5] arranges in a multi-hop topology. The Constrained Application Protocol (CoAP) [6] offers a lightweight alternative to HTTP while running over UDP, or DTLS [7] for session security. This set of solutions extends the host-centric end-to-end paradigm of the Internet to the embedded world and puts IPv6 in place for loosely joining the *things*.

Doubts arose whether host-to-host sessions are the appropriate approach in these disruption-prone environments of (wireless) things, and the data-centric nature at the Internet edge called for rethinking the current IoT architecture [8]. ICN networks [9] have been identified as promising candidate networks for a future IoT. Name-based routing and in-network caching as contributed by Named Data Networking (NDN) [10] bear the potential to increase robustness of application scenarios in regimes of low reliability and reduced infrastructure (*e.g.,* without DNS). The quest for the best solution remains open. Rather little is known about the differences and commonalities when deploying the varying protocols in the wild. This surprisingly unsatisfying state of the art motivates us to implement, deploy, and thoroughly analyze the different protocols in typical use cases and scenarios for the constrained IIoT.

The main contributions of this paper shed light on a systematic and comprehensive understanding of protocol design for the IIoT. In detail:

1) We characterize important industrial use cases and summarize requirements, backed up by field experiences of the safety-critical industry. These requirements serve our evaluations and may guide future analyses.
2) We perform a thorough comparative analysis based on extensive real-world experiments, including dense

C. Gündoğan, P. Kietzmann, and T.C. Schmidt are with Hamburg University of Applied Sciences, Germany (e-mail: {cenk.guendogan, peter.kietzmann, t.schmidt}@haw-hamburg).

M. Lenders, H. Petersen, M. Wählisch are with Freie Universität Berlin, Germany, (e-mail: {m.lenders, hauke.petersen, m.waehlisch}@fu-berlin.de).

M. Frey, F. Shzu-Juraschek are with Safety io, Germany, (e-mail:{Michael.Frey, Felix.Juraschek}@safetyio.com).

scenarios of 50 constrained nodes. We consider three common protocol families, CoAP, MQTT[1], and NDN,
3) We make our implementations publicly available and thus provide an evaluation framework for protocol assessment in the IIoT.

This paper extends our previous work from ICN 2018 [11] by refocusing on the industrial use cases and by adding many experimental analyses tailored to the industrial requirements. Our analysis revealed significant differences in protocol behavior without an overall winner: The challenging multi-hop domain is best mastered by the ICN protocols, while MQTT-SN proved most resistant against cross-traffic from coexisting networks on the price of bursty occupation of network resources. Lightweight publish-subscribe protocols such as CoAP observe and MQTT-SN operate fastest and most versatile under relaxed wireless conditions.

The remainder of this paper is structured as follows. Section II introduces safety-critical use cases and derives requirements for networking. The following Section III summarizes the related work on key protocol concepts along with a qualitative comparison. Section IV explains our implementations and experimental setup. We present our measurements with a special focus on the impact of single- versus multi-hop topologies and uncontrolled side traffic in Sections V and VI, respectively. In Section VII, we revisit the related work on protocol performance and conclude in Section VIII.

II. Industrial IoT Use Cases

The Industrial Internet of Things (IIoT) revolutionizes how processes in industrial environments are controlled. It makes use of local data aggregation, processing on the edge, and cloud computing to refine and optimize process controls. Here, infrastructure such as sensors and actuators are interconnected. Applications range from aggregating locally stored log data (reporting) to sensing and raising alarms if thresholds are undercut or exceeded (monitoring) and even intervening processes with regulating actuators (controlling) [12]. The aggregation of log data has to be reliable and secure. Reliable in a sense that all existing data needs to be transferred to a designated device, secure within respect to the integrity and authenticity of the data. While the sensing and propagation of non-critical data in-network has modest timing requirements, critical data such as alarms and control commands need to be forwarded and disseminated to relevant parties with low latency.

Among others, use cases include monitoring and reporting of environmental and vital data of workers in harsh environments (*e.g.,* early responders), process regulation in manufacturing industries (*e.g.,* chemicals, gas, oil, minerals), and even factory automation using with control of robots (*e.g.,* assembly lines) [13]. An overview of the communication flows as typical for industrial environments is visualized in Figure 1.

Safety-critical Environments. Safety-critical environments clearly benefit from the Industry 4.0 paradigm, *i.e.,* networking

[1]We use MQTT-SN, the UDP-adapted version in the MQTT family, because TCP is inappropriate for the constrained IIoT

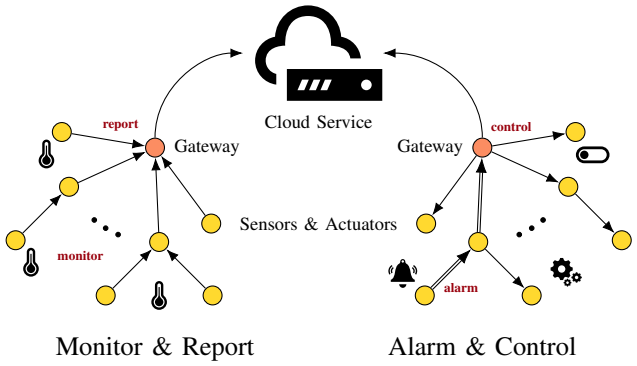


Fig. 1. Communication flows in IIoT environments.

the control components, because here advanced communication interfaces do not only improve manufacturing processes but may also help to save lives. Concrete examples are industrial processing plants or refineries. Typically, a mix of personal mobile and fixed gas detectors are used to sense the environment for possible leaks of hazardous or combustible gases. Often multiple teams of workers perform maintenance tasks in designated areas, in which every worker is equipped with a gas detector. In addition, fixed gas detectors are deployed on critical infrastructure, which support the maintenance tasks of workers. Detection of a dangerous level of gas switches the gas detectors to alarm mode. In a networked scenario, each detector sends a message to a centralized safety monitoring application that runs either locally or in the (edge) cloud. Based on new alarm information, the safety manager decides whether to preemptively evacuate close-by areas.

Industrial Control Systems. Control systems are widely deployed in industrial process automation, where they continuously monitor flows, and in factory automation, where they mainly deal with discrete on/off signals of machines like robots. Continuous monitoring periodically transmits process values and directly adjusts control of actuators such as valves or pumps in a closed loop. In contrast, discrete control signals are event driven (e.g., generated from a relay after a robot action) and require individual reactions, which are not stabilized by corrective periodic updates. Control events may be critical and consequently more sensitive to signal delays or losses.

Deployment of sensors and actuators in industrial production environments is likewise harsh. Plants often undergo unpredictable variations in the environment (e.g., temperature, humidity, vibrations), in the radio regime (e.g., cross traffic, reflections from moving metal objects, steam emittance from machines), and energy-wise. Many field devices operate on batteries and may need to survive periods from days to months between recharging and general overhauls.

Requirements. From the networking perspective, the following requirements have to be satisfied to fully support these industrial safety use cases. The most important requirement is latency for alarm messages from the detector to the safety application and for evacuation requests in the reverse direction. Being able to react quickly to safety

relevant incidents is crucial to contain and resolve dangerous conditions. The ANSI/ISA-100.11a-2011 standard [14] defines latency requirements for three traffic categories in industrial process automation applications:

1) *Safety* traffic indicates emergency and requires a maximum of 10 ms delay in a deterministic fashion.
2) *Control* traffic is often but not always critical and depends on its application context, latencies between 10 and 100 ms are sufficient.
3) *Monitoring* traffic is used for maintenance and should deliver messages within 100 ms on average.

Additionally, lost messages may lead to undetected alarms in the safety monitoring software and, hence, a *high reliability* is crucial.

When not in alarm mode, detectors log their sensor readings on the device and send their status once per minute. This frequency increases when a detector changes to alarm state, since regulations stipulate that the sensor readings need to be logged at least once per second. Those logs are required for any investigation following up the particular gas alarm incident. Thus, it is desirable to *send data at very low frequency* to the centralized safety application.

From an operational perspective, the network architecture should allow for the deployment of a flexible ecosystem, which enables private as well as open networks.

Challenges. Meeting these requirements is challenging in harsh industrial environments, where time-slotting traffic schedules are difficult to deploy. Workers are constantly moving, and path-loss and shadowing effects appear due to the massive amounts of steel used in processing plants. In addition, there may be uncoordinated side channel traffic initiated by co-located systems from different manufacturers, which is particularly harmful for synchronized communication channels as defined in IEEE 802.15.4e (TSCH) [4] deployments [15], [16]. In case of larger incidents, in which several hundred or thousand detectors send alarm notifications, coping with network traffic is even more challenging. And finally, some industrial areas are so remote that network coverage provided by technologies such as cellular is very poor or non-existing.

On the upside, monitoring the complete gas detector status typically fits in less than one kilobyte of data. Thus, the required available data rate is very low.

Potentials of 5G. A key building block for a successful IIoT is 5G [2]. Massive machine type communication (mMTC) provides a narrowband Internet access for sensing, actuating, and monitoring devices. The ultra-reliable low latency communication (URLLC) in 5G will provide sub-millisecond latency communication, which is essential for dedicated devices in process control. Additionally, allowing industrial customers to operate their private 5G-based networks provides the chance to close coverage gaps in remote areas. These private networks can then be inter-connected with a mobile carriers network. Finally, 5G opens the scene for a data-centric network core, which may help to increase reliability in constrained and lossy environments.

Having a promising network access architecture such as 5G in place still requires efficient protocols on top. The current IIoT ecosystem proposes several competing solutions. These protocols require careful evaluation with respect to resource allocation, convergence problems, and coexistence scenarios, in particular in the context of a safety-critical Industry 4.0.

III. Networking Protocols for Industry 4.0

Domain-specific protocols in the IIoT include Zigbee, ISA100.11a, and WirelessHART [13], [17], all of which specify a full protocol stack which can be configured to application requirements. This is done by a centralized instance, usually called a network manager. The network- and transport layers deal with IP connectivity on a backbone router whereas routing between constrained devices is implemented in a proprietary fashion directly on top of the MAC layer.

Standard IoT networking protocols to handle massive volumes of heterogeneous data flows are CoAP and MQTT on the application layer in the current Internet, and information-centric (or data-centric) networking for the next generation IIoT. The latter provides higher layer services known from the application layer, such as naming and caching, directly on top of the data link layer. In this section, we briefly give technical background to common link technologies in the IIoT, and provide a qualitative comparison of the core protocols CoAP, MQTT and ICN.

A. Common Link Layers for the IIoT

Industrial protocols to handle data flows of sensors and actuators heavily rely on the MAC at its link layer, which we briefly discuss here. The popular 802.15.4 family is a characteristic example of lossy local area wireless transmission at minimal energy. We base our experimental work on 802.15.4 with non-slotted media access to provide robust transmissions and neutral performance impact, for the absence of time schedules.

IEEE 802.15.4-based technologies.

Many short range wireless solutions in the IIoT are built on IEEE 802.15.4, which specifies low-power and low-rate physical layers and media access control. Prominent examples are Zigbee, ISA100.11a, ore WirelessHART. The PHY in most deployments operates on the 2.4 GHz band and applies a simple O-QPSK (Quadrature Phase Shift Keying) modulation. Symbols are spread in the code domain to operate on a direct-sequence spread spectrum (DSSS). This increases resistance against narrow-band interference.

We distinguish two classes of media access with this technology: (i) time-slotted and (ii) non-slotted multiple access. The former reduces energy consumption, though, its performance is heavily affected by the scheduling logic upfront. Furthermore, network synchronization is susceptible to interference. In contrast, non-slotted access omits scheduling and exploits carrier sensing to avoid collisions.

Wireless media is susceptible to eavesdropping, and security between neighbored nodes is provided by the 802.15.4 MAC. Hence, higher layer security is still required to achieve security on data domain. 802.15.4 specifies eight levels of protection

which reflect increasing security strengths to achieve data privacy, integrity, and authenticity. Data encryption and message integrity codes utilize AES with 128 Bit keys, though, provisioning of keys between peers needs to be handled by the upper layer, or manually during deployment. In addition, access control lists exclude frames that are received from untrusted nodes and hence, could be malicious. It is noteworthy, though, that bare 802.15.4 is still vulnerable to a number of attacks [18], [19].

All three standards mentioned above utilize the time-slotted channel hopping mode of the IEEE 802.15.4e specification to guarantee link resources. This type of time- and frequency multiplex requires coordination among nodes to synchronize to a schedule, and to grant resource access. Hence, it adds signaling overhead, especially for sporadic and asynchronous data. The slot mode, however, enables device sleep cycles to save energy. The IETF adopted 6TiSCH [20], [21] as an open standard solution that bases on the above mentioned protocols and enables IPv6 connectivity on constrained nodes themselves. Due to central coordination and susceptibility to side-channel interference [15], [22], however, TiSCH-type link layers do not meet the use cases of uncoordinated deployment in harsh industrial environments. We therefore base our experimental work on the contention-based and grant-free CSMA/CA mode of IEEE802.15.4 and concentrate on the performance impacts of the higher layers.

Novel, non-orthogonal technologies. Orthogonal access schemes like 802.15.4 as presented above, are key to current wireless systems, however, the orthogonality criterion limits the number of users. Consequently, mMTC platforms advance in modulation and multiplexing by introducing non-orthogonal schemes to the space, time, frequency, or code domain [23], [24]. This allows for resource overloading to extend the number of simultaneous users but also increases receiver complexity. Sparse code multiple access (SCMA) is a core technique in 5G systems which operates in the code domain to enable overloading. SCMA maps data-streams to non-orthogonal code streams. Codewords of multiple SCMA-layers are combined and transmitted over OFDMA (orthogonal frequency-division multiple access), a multi-carrier technique with time slotted access. Space division is achieved by traditional cell clustering and advanced with antenna beamforming to reduce cell overlap, and thus, to increase resource re-utilization. Hence, 5G extends media access in four dimensions: code, frequency, time, and space.

B. Common Application Layer Protocols for the IIoT

The IETF solution, CoAP. The Constrained Application Protocol (CoAP) [6] aims for replacing HTTP to enable M2M communication between constrained nodes. In contrast to HTTP, CoAP is able to run on top of UDP and introduces a lean transactional messaging layer to compensate for the connectionless transport. CoAP provides a more compact header structure than HTTP. It currently supports three communication primitives: (*i*) pull, (*ii*) push, and (*iii*) observe. Pull implements the common request response communication pattern. However, as IoT scenarios also include the pro-active communication of unscheduled state changes, CoAP was extended to support pushing new events to its peers. Still, this does not allow for publish-subscribe scenarios when producer and consumer are decoupled in time and data is not yet available at the request. The support for delayed data delivery in publish-subscribe was specified in CoAP observe [25]. Here, clients can signal interest in observing data, which implies that a CoAP server delivers data as soon as available and maintains state until clients explicitly unsubscribe. The default approach to reinforce communication channels in CoAP deployments is to use (datagram) transport layer security (D)TLS [7], [26]. OSCORE [27] is a recent addendum to the CoAP specification and allows for securing content objects on the application layer, in addition to any transport protection.

CoAP is the IETF standard for implementing data transfer on the application layer in the future Industrial Internet of Things. Currently, several implementations exist, as well as early adoption in a few selected products and deployments.

The well-deployed solution, MQTT. The Message Queue Telemetry Transport (MQTT) [1] was designed as a publish-subscribe messaging protocol between clients and brokers. Clients can publish content, subscribe to content, or both. Servers (commonly called *broker*) distribute messages between publishing and subscribing clients. It is worth noting that the protocol is symmetric: Clients as well as brokers can be sender and receiver when MQTT delivers application messages.

MQTT is considered a lightweight protocol for two reasons. First, it provides a lean header structure, which reduces packet parsing and makes it suitable for constrained devices with low energy resources. Second, it is easy to implement. In its simplest form, MQTT offloads reliability support completely onto TCP.

To provide flexible Quality of Service on top of the underlying transport, MQTT defines three QoS levels. *QoS 0* implements unacknowledged data transfer. An MQTT receiver gets a message at most once, depending on the capabilities of the underlying network, as there is no retransmission on the application layer. *QoS 1* guarantees that a message is delivered at least once. Based on timeouts, an MQTT sender will retransmit application messages when an acknowledgment is missing. *QoS 2* ensures that a message is received exactly once, to avoid packet loss or processing of duplicates at the MQTT receiver side. This requires a two-step acknowledgment process and more state at both sides.

To adapt MQTT to constrained networks which are based on low data rates and very small packet lengths such as in 802.15.4, MQTT-SN [28] is specified. Header complexity is reduced by replacing topic strings with topic IDs, to identify content. In contrast to MQTT, MQTT-SN is able to run on top of UDP. It still supports all QoS levels but does not inherit any reliability property from the transport layer.

MQTT provides optional header fields during the establishment of connections to authenticate with a broker, but most other responsibilities, such as encrypting and authenticating published data, are relayed to the application. The transport is commonly protected using transport layer security (TLS) for the TCP-based MQTT, and the datagram variant DTLS for MQTT-SN. The specification provides implementation

TABLE I
COMPARISON OF COAP, MQTT, AND ICN PROTOCOLS. COAP AND MQTT SUPPORT RELIABILITY ONLY IN CONFIRMABLE MODE (C) AND QOS LEVELS 1 AND 2 (Q1, Q2).

	Current IoT Protocols					**ICN Protocols**	
	CoAP			MQTT	MQTT-SN	NDN	HoPP
	PUT	GET	Observe				
Transport	UDP	UDP	UDP	TCP	UDP	n/a	n/a
Pub/Sub	✘	✘	✓	✓	✓	✘	✓
Push	✓	✘	✓	✓	✓	✘	✘
Pull	✘	✓	✘	✘	✘	✓	✓
Flow Control	✘	✘	✘	✓	✘	✓	✓
Reliability	(c)	(c)	✘	(Q1, Q2)	(Q1, Q2)	✓	✓
Security Mechanism	transport / content object	transport / content object	transport / content object	transport	transport	content object	content object
End-to-end Protection	(✓)	(✓)	(✓)	✘	✘	✓	✓

notes and guidance for a secured deployment in the protocol specification [1, Section 5].

C. Upcoming Data-centric Networking Layers

Information-centric networking (ICN) implements the vision of a native data-centric Internet. The most active approach is named-data networking (NDN). The core **NDN** protocol [10] combines name-based routing with stateful forwarding to deploy a request response scheme on the network layer. Any consumer can request named data using so-called Interest messages, which are forwarded towards publishers. Data is subsequently delivered along a trail of reverse path forwarding states, starting either from the original publisher or the first in-network cache that can provide the requested data. As an important feature, data will only be delivered to those who requested the data. This means that data must be (individually) named at the Interest request and that yet unavailable data requires repeating Interests until the application receives the data. Due to the comprehensive use of on-path caches and the stateful forwarding fabric, the concept of endpoints becomes negligible for NDN deployments. Thus, these regimes allow for an orthogonal approach of delivering autonomously verifiable content objects independently of location and communication endpoints.

Several publish-subscribe extensions have been proposed for NDN [29]–[31] to provide further decoupling of consumers and data sources. **HoP and Pull (HoPP)** [31] is a lightweight variant we previously developed to provide a publish-subscribe system for constrained IIoT deployments based on ICN/NDN principles. A constrained publisher announces a name towards a content proxy to trigger content requests and to replicate the data towards a content proxy (or broker). Forwarding nodes on the path between publisher and content proxy hop-wise request content for this name by using common Interest and data messages. A content subscriber in HoPP behaves almost like any content requester in NDN and issues a regular Interest request towards the content proxy. However, in contrast to NDN (*i*) a subscriber cannot extract content names from its forwarding information base (FIB), since FIBs only contain default routes [32], but uses application-specific topic tables instead; (*ii*) it does not expect an immediate reply, but issues Interests with extended lifetimes. HoPP enables rapid communication of unscheduled data events. It operates at a similar timescale as push protocols without actually pushing data.

D. Qualitative Comparison of Protocols

Key properties of the three protocol families CoAP, MQTT, and NDN and its variants are compared in TABLE I. Specialized properties of the different approaches become apparent: Every protocol variant features distinct capabilities. Notably in the IoT, where TCP (aka generic MQTT) is unavailable, the pull-based NDN and NDN-HoPP are the only protocols admitting flow control and reliability as a generic service. Further, low-power deployments show a growing demand for application gateways to perform protocol conversions and changing the transport, *e.g.,* from UDP to TCP. These operations naturally break the end-to-end principle [33] at gateways, terminate any transport security, and therefore render the communication between constrained IoT devices and cloud services vulnerable to interception attacks [34], [35]. The NDN family of protocols and CoAP with OSCORE protection can guarantee security properties to remain intact beyond protocol conversions [36]. In addition, content object security enables multicast and multi-homing capabilities for the IoT, while these are hardly feasible to deploy with transport protection due to the tight endpoint binding. This especially affects setups that experience device mobility and frequent network disruptions.

To give a first estimate of the different protocol complexities, we compare the sizes of the message types for each protocol in Figure 2. Most of the protocols need nearly the same amount of data. CoAP observe (CoAP OBS) exhibits the lowest complexity but does not acknowledge. A single registration is sufficient to receive subsequent data published under the same name. HoPP, on the other hand, introduces overall the largest packet size as it introduces name advertising on the data plane. We elaborate the scenario and give a comprehensive performance evaluation in the next sections.

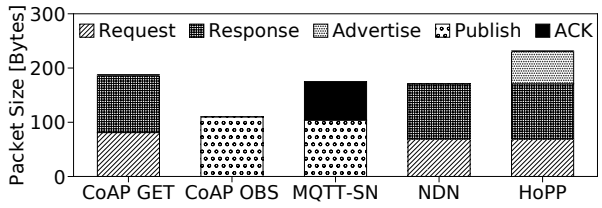


Fig. 2. Packet sizes in bytes for each protocol.

IV. An Environment for Assessing Industrial M2M Networks

Common deployments in the IIoT consist of stub networks that are single-hop in areas of dense infrastructure, but may also be multi-hop in widespread facilities such as oil refineries or platforms. Traffic flows from or to the edge nodes in three patterns: (*i*) scheduled periodic sensor readings, (*ii*) unscheduled and uncoordinated data updates, or (*iii*) on demand notifications or alerting. It is worth noting that the different protocol properties (e.g., pub-sub versus request-response) meet these alternating demands differently well. In the following, we present a testing environment consisting of software, a real-world testbed, and relevant scenarios that approximates the characteristics of massive M2M networks for embedded devices.

A. Software Platforms

On the constrained nodes, all of our experiments are based on the RIOT operating system [37] version `2018.01`. To analyze CoAP, MQTT-SN, and NDN we use `gCoAP`, `Asymcute`, and `CCN-lite` respectively. All three protocol implementations are part of the common RIOT release and thus reflect typical software components used in low-end IoT scenarios.

Brokers or gateways are deployed on Linux systems within the testbed infrastructure. To support an MQTT-SN broker and a CoAP observe client, we used `aiocoap` version 0.3 and `mosquitto.rsmb` version 1.3.0.2. Both are popular open source implementations in this context.

B. Testbed

We conduct our experiments in the FIT IoT-LAB[2] testbed. The hardware platform consists of typical class 2 devices [38] and features an ARM Cortex-M3 MCU with 64 kB of RAM and 512 kB of ROM. Each device is equipped with an Atmel AT86RF231 transceiver to operate an IEEE 802.15.4 radio. The gateway runs on a Cortex-A8 node, which is more powerful than the M3 edge nodes. Every node in the testbed is monitored by a control node which allows for parallel radio sniffing without misusing transceivers of M3 devices.

The testbed provides access to several sites with varying properties. We perform our experiments on different sites, to analyze single-hop as well as multi-hop scenarios.

Single-hop topology The *Paris* site consists of approximately 70 nodes, which are within the same radio range. We choose two arbitrary nodes and run all single-hop experiments on them. One node is a content producer, the other node acts as consumer (gateway/broker).

Multi-hop topology The *Grenoble* site consists of approximately 350 nodes spread evenly in the Inria Grenoble building. We choose 50 M3 nodes (low-end device) and one A8 node (gateway/broker) arbitrarily and run all multi-hop experiments on them. All low-end devices operate as content producers. In our CoAP and MQTT experiments, we use RPL to build and maintain the routing topology across all nodes. For NDN-based experiments we build analogous tree topologies. Typical path lengths are four to six hops.

Two-hop topology with cross-traffic We choose three M3 nodes that are arranged in a line topology within the *Grenoble* site. One node acts as a consumer, another node serves as a producer and the last node is a forwarder in between. Additionally, we deploy a fourth node acting as a cross-traffic generator in the vicinity of our forwarder.

C. Scenarios and Parameters

We align all experiments with respect to the configurations of retransmissions and timeouts to ensure comparability among protocols. All protocols employ the same retransmission strategy: In case of failures, each node waits 2 seconds before retransmitting the original application or control data. For NDN and HoPP, retransmissions are performed hop-by-hop, while CoAP and MQTT retransmit from end to end. At most 4 retransmissions will occur for each data item. Interest lifetimes are configured to 10 seconds for NDN based protocols to limit PIT memory consumption. We repeat each experiment 1,000 times.

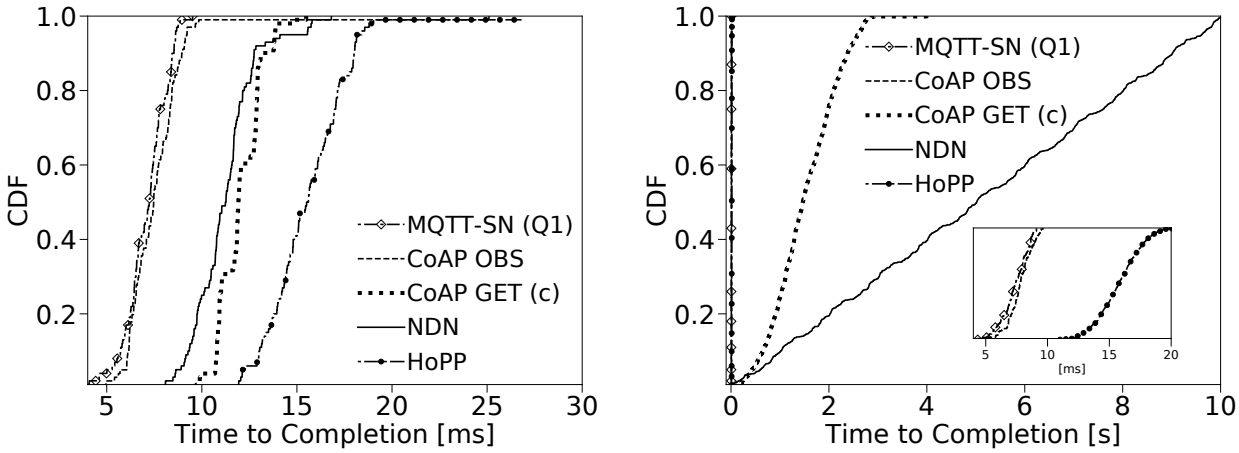
To accommodate all 50 nodes in the routing topology, the FIB sizes have been adjusted accordingly on each constrained node. For CoAP and MQTT, this translates in our IPv6 scenario to a FIB size of 50 entries with roughly 32 bytes each. In our NDN scenarios, each node owns a unique prefix of the form $/\rho_i$ with a length of `24` bytes. The next-hop face of each FIB entry points to the `8`-byte IEEE 802.15.4 link-layer address. In total, this setup yields comparable size requirements for all scenarios.

In the NDN scenarios, we use unique content names prefixed by $/\rho_i$ with incremental local packet counters. CoAP works without unique names but uses common URIs. The MQTT-SN protocols register a common topic name, similar to CoAP, and publish under a unique topic ID thereafter. In all scenarios, the data is of the same JSON format consisting of a unique identifier and a sensor value attribute. These short messages can be accommodated by the link layer and do not require fragmentation. It is noteworthy that we neither apply header compression in the IP [39] nor in the NDN world [40].

V. The Impact of Topology in Massive Deployment

The objective of this work is to quantify the impact of network protocols on IIoT communication systems. With this goal in mind, we deploy the different publish-subscribe and request-response protocols in the same physical environment

[2]http://www.iot-lab.info/



(a) Scheduled publishing at 1 s interval

(b) Unscheduled random publishing in 1s . . . 3s

Fig. 3. Time to content arrival in a single-hop topology.

and compare their operational properties as well as their performance results. Evaluation metrics focus on reliability and timeliness of the data delivery, which are critical in the low power lossy environment of these systems. Additionally, we study link stress and resource efficiency of the constrained data flows. We start our analysis by comparing single- versus multi-hop topologies.

A. Single-Hop Topology

Protocol performances are first evaluated in a single-hop topology at the Paris testbed of IoT-LAB. In agreement with the requirements of our industrial use case, we perform a periodically scheduled publishing at every second, and a randomized, unscheduled publishing. We measure the time until content arrives at the consumer. The results are summarized in CDFs as functions of packet transfer time, see Fig. 3.

In the case of scheduled traffic, all protocols successfully deliver data packets within short, similar times as shown in Fig. 3(a). Lightly visible steps in the CDFs indicate retransmissions on layer 2, which occur on the same timescale of milliseconds. Naturally, the protocols that push data (MQTT, CoAP OBS) react quicker than request-response schemes. As a pull-based publish-subscribe scheme, HoPP performs slowest, as it initiates hop-wise data transfers on request.

Our second evaluation addresses the common IoT use case of publishing data at irregular intervals. This is the typical pattern for observing third party actions (*e.g.,* alarms), or largely uncoordinated sensing environments. The publish-subscribe protocols naturally serve these application needs. We quantify the behavior of the request-based protocols in practice and chose the moderate setting of publishing content every two seconds on average. Publishing is uniformly distributed in the interval of $[1\,s,\ldots 3\,s]$. The protocols CoAP and NDN request the content periodically every second so that updates are not lost.

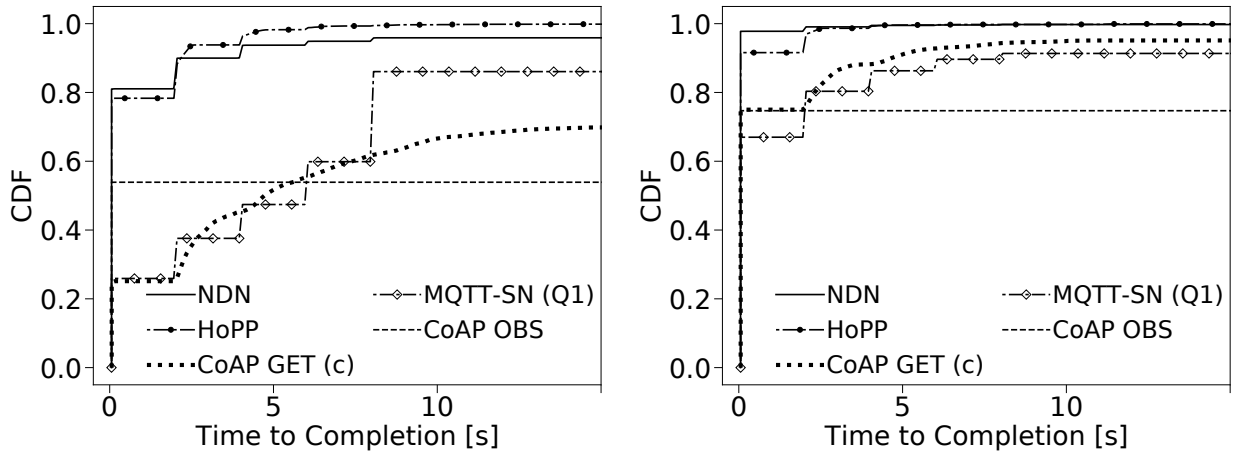
Fig. 3(b) visualizes content delivery times for unscheduled publishing and reveal a diverse picture. CoAP GET and NDN now operate on a timescale of seconds, while the publish-subscribe protocols continues to complete in the unaltered range of $10\,ms$ without additional protocol operations – the unsurprising outcome of content triggers. CoAP requests content using a common name with the result of likely duplicate content transmissions. On average, CoAP needs two requests to retrieve fresh content with the expected average delay of $\approx 2\,s$ and a corresponding polling overhead of 200 %, see Fig. 3(b). In contrast, NDN admits lower overhead, as Interests are locally managed at the PIT and only retransmitted after state timeout. Issuing Interests at a higher rate than content arrival, however, leads to an accumulation of open states in the PIT. As resources on the constrained nodes are tightly bound, the PIT limits are quickly reached and can be only met by either *discarding* newly arriving Interests, or by *overwriting pending Interest state*. Both countermeasures delay content delivery, as can be seen in Fig. 3(b).

B. Multi-Hop Topology

We now consider the more challenging use case of mixed communication in multi-hop topologies: 50 nodes exchange content that is published every 5 or 30 seconds in an uncoordinated manner. Repeated experiments were performed on the Grenoble testbed with tree topologies of routing depths varying from four to six hops.

First, we examine the temporal distributions from content publishing to arrival in analogy to the single-hop cases. Fig. 4 combines the results for all protocols, as well as both publishing rates. The overall results reveal a much slower and less reliable protocol behavior than could be expected from the single-hop values in Fig. 3. Graphs reflect the common experience in low power multi-hop environments that interferences and individual error probabilities accumulate in a destructive manner.

The IP-based protocols, which operate in an end-to-end paradigm, now all fail in delivering data, the publish-subscribe protocols CoAP OBS and MQTT-SN representing the lower end. Widespread temporal distributions indicate repeated retransmissions on the network layer that operate on the scale of many seconds and still cannot compensate losses. In contrast, the hop-by-hop nature of the ICN protocols enfolds its



(a) 5 s publishing interval

(b) 30 s publishing interval

Fig. 4. Time to content arrival in multi-hop topologies of 50 nodes for publish-subscribe and request-response protocols at different publishing intervals.

robustness in these harsh environments. The publish-subscribe protocol HoPP quickly reaches 100 % success in data transfer – 80 % (Fig. 4(a)) resp. 95 % (Fig. 4(b)) of data units arrive within milliseconds and without any network layer retransmission. The performance of the plain NDN also shows decent results both in promptness and reliability, even though 5 % of data chunks remain lost in the fast publishing scenario of 5 s.

Second, we focus on the link utilization. We measure all individual paths that each unique data packet traveled on its destination from source to sink, and contrast the results with the corresponding shortest possible path. Results are visualized as scatterplots in Fig. 5. Each dot represents one or several events, the dot size is drawn proportionally to event multiplicities. Solid lines indicate the shortest paths, while events left of the line represent failures (traversal shorter than the shortest path). Right of the solid diagonal retransmissions are counted.

The ideal protocol performance is situated on the diagonal line with all data traversing each link only once on the shortest path. This ideal behavior is most closely approximated by the NDN core and the NDN-HoPP protocols. A largely contrasting performance can be seen from the reliable IP protocols MQTT-SN (Q1), which admits huge numbers of retransmissions. These retransmissions stress an exhausted link even further and stimulate cascading failures. The CoAP protocol variants behave more network friendly, thereby accumulating loss in a polite fashion.

We further question the details of packet loss and count the transmission failures on each link during the experiment. Fig. 6 displays the number of packets lost in one minute as a function of time and hop distance from the gateway. Note that in this analysis every packet lost on some link is counted, no matter whether the retransmission mechanisms on the different layers can compensate this loss. An overall successful packet transfer in this analysis can thus account for many loss events on intermediate links. Frequent losses indicate a less effective link utilization by the network protocol.

It is common for this convergecast scenario that loss intensity increases toward the gateway, which serves as the root of the routing tree. Here packets accumulate on the last hops, why link exhaustion, collisions, and buffer drops increase. The effective success rate of packet traversal is largely influenced by the flow properties (*i.e.,* bursts versus balanced flows) as shaped by the networking protocol. In this, the protocol behaviors largely differ and lead to diverging results. The ICN protocols NDN and HoPP in Fig. 6 show a more random distribution of small losses, which is typical for wireless interference and can be compensated by local retransmissions. In contrast, the IP-based protocols all suffer from more intense losses close to the gateway—loss of IP packets exceeds ICN loss by factors between 10 and 100. Only CoAP OBS looses moderately, because CoAP retransmissions are not active in this protocol variant and the total number of packets remains lower.

Compared to the confirmable CoAP GET configuration, MQTT-SN exhibits less loss events on the links farther away from the source, because of its more compact packet encoding. Extreme loss values show up at the source for MQTT-SN, however, due to its uncoordinated, bursty retransmissions. These effects amplify in the multi-hop tree topology as the total network traffic accumulates towards the few links that directly connect to the gateway node. This explains the details behind the large transmission numbers seen in Fig. 5.

Next, we comparatively examine the nodal energy consumption as a function of time throughout an experiment duration of ≈60 minutes for each deployment in our protocol selection. In the typical IoT scenario of acquiring and distributing sensor readings, energy expenditures due to computational efforts usually remain within tolerable limits. Radio activities, on the other hand, dominantly drive energy demands when receiving and transmitting data over the air. To concentrate on power expenses based on protocol characteristics, such as packet sizes and the quality of corrective actions, we only measure energy levels for actual radio operations. Consequently, we disregard expenses due to actively listening on the radio in our calculations, since this part of the equation decreases substantially in proper deployments with correct duty cycling

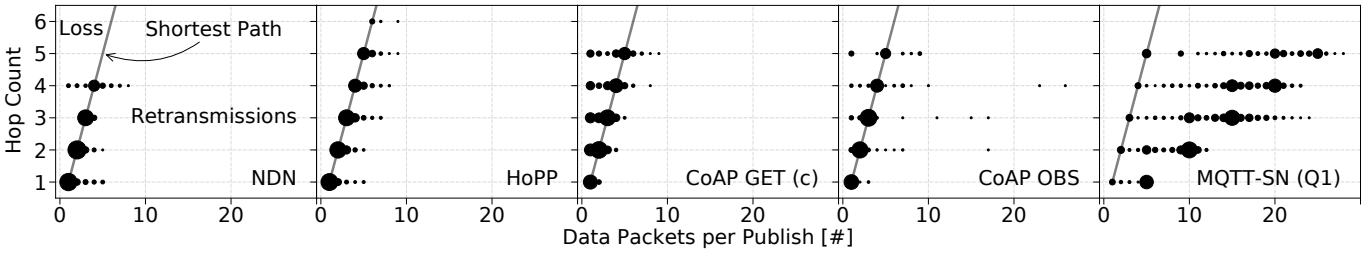


Fig. 5. Link traversal vs. shortest path for a 30 s publishing interval. The scatterplots reveal the link stress with dot sizes proportional to event multiplicity.

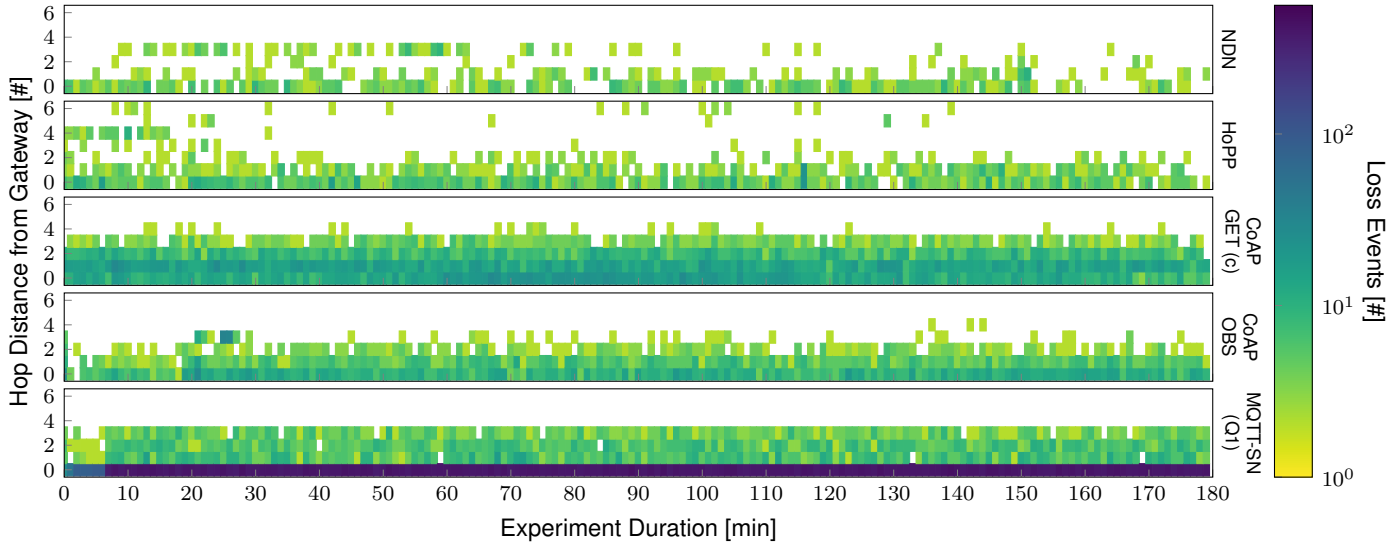


Fig. 6. Loss count at links as a function of experiment time and hop distance. Cells show the loss intensity per minute for a 30 s publishing interval.

and utilizing low-power modes. We obtain the power consumption levels for transmitting and receiving from the Atmel AT86RF231 [41] data sheet and convert nodal packet statistics into appropriate energy expenditures.

TABLE II compiles the statistical key properties for the nodal energy expenses of the 50 nodes in our multi-hop setup with a publishing interval of 30 s. Principally, maximum values represent energy levels of gateway nodes, since packets naturally accumulate there due to the convergecast setup. The 25% (Q1) and 75% (Q3) quartiles roughly illustrate the energy distributions. We note that nodes closer to the gateway are much more engaged with forwarding duties and experience additional radio activities when compared to leaf nodes. Thus, these nodes generally position towards the higher end of the distribution.

The average consumption for a single node greatly varies between the selected configurations, but agrees with our previous conclusions in Fig. 5 and Fig. 6. CoAP OBS displays the lowest average expense with ≈55 mJ per node, which is expected due its push-based nature and the lack of retransmissions. MQTT-SN presents another extreme: the excessive amount of corrective actions, especially at the gateway—see the elevated maximum in TABLE II—leads to an average expense that is fourfold. CoAP GET situates between both configurations with an average of ≈152 mJ. NDN operates reliably throughout the experiment (see Fig. 4(b)) with a minimal number of packets in the network. Shortened retransmission paths with on-path caching are the key protocol features of NDN to reduce overall energy expenditures down to an average of ≈99 mJ and still maintain distinct success rates. Since HoPP counts a link-local signaling overhead for each published data, the total power consumption slightly elevate.

Last, we dive deeper into the flow balance of the different protocols and evaluate its effective data goodputs during various content publishing experiments. Fig. 7 summarizes the results. We display the distribution of goodput from the different experiments in box plots and compare to the theoretical optimum (lines). Time series of data goodput further reveal the flow behavior as displayed in the lower row of the figure.

Clearly, HoPP admits the most evenly balanced flows and shows nearly optimal goodput values, closely followed by NDN. All other flow performances fluctuate with some tendency of instability when approaching its full transmission speed. Some IP-based flows in MQTT-SN and CoAP drop to lower delivery rates which is dominantly caused by slow repeated end-to-end retransmission. Multi-hop retransmissions in this error-prone regime tend to cause additional interferences and accumulate transmission errors. As a consequence, protocols operate at reduced efficiency – for CoAP OBS protocol performance drops down to 50 %. The overall results show that the absence of flow control as in UDP/IP–based protocols make protocols fragile. Hop-wise retransmission

Protocol	μ [mJ]	σ [mJ]	min [mJ]	Q1 [mJ]	median [mJ]	Q3 [mJ]	max [mJ]	sum [mJ]
NDN	98.99	213.96	23.66	23.66	23.88	70.98	1,243.54	4,949.50
HoPP	167.33	271.50	34.69	37.55	44.87	158.37	1,494.29	8,534.27
CoAP GET (c)	151.61	293.72	25.62	27.94	29.54	82.26	1,411.53	7,732.13
CoAP OBS	55.78	89.66	10.59	12.88	20.17	42.92	371.84	2,844.80
MQTT-SN (Q1)	245.66	394.63	65.61	68.66	74.91	183.10	1,915.61	12,529.12

TABLE II
STATISTICAL KEY PROPERTIES OF NODAL ENERGY EXPENDITIURES W.R.T. RADIO TRANSCEIVER OPERATIONS, *i.e.,* ACTIVELY SENDING AND RECEIVING. VALUES CALCULATE OVER THE EXPERIMENT DURATION FOR OUR PROTOCOL SELECTION CONFIGURED WITH A 30 S PUBLISHING INTERVAL. Q1 AND Q3 REPRESENT THE FIRST (25%) AND THIRD (75%) QUARTILE, RESPECTIVELY.

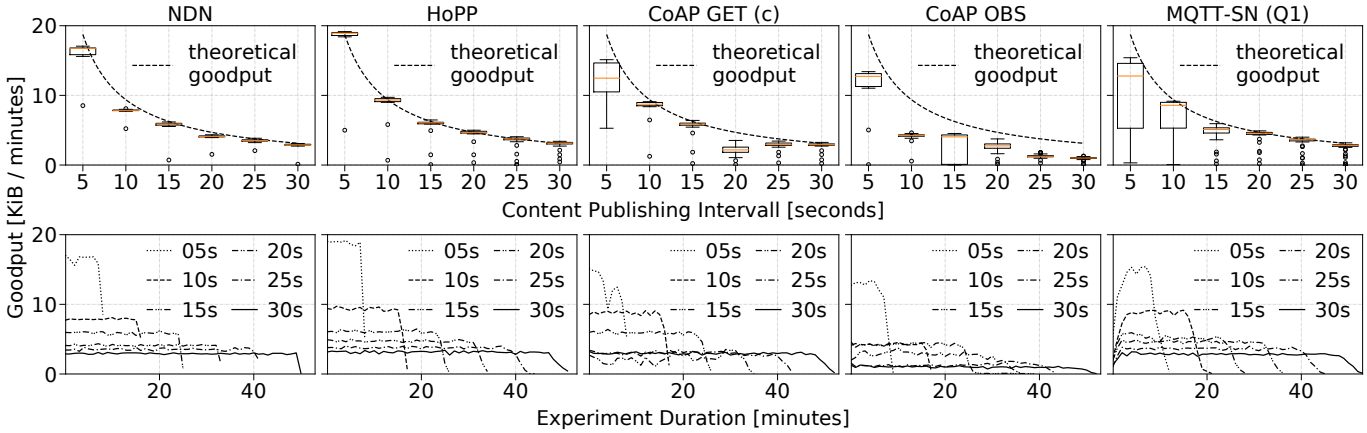


Fig. 7. Goodput summary and flow evolution for all protocols at different publishing intervals.

management as applicable in NDN and HoPP re-balances flows and explicitly demonstrates its benefits for the IIoT instead.

VI. THE IMPACT OF COEXISTING WIRELESS NODES

We continue our protocol analysis by investigating the case of uncontrolled disturbances. In unshielded environments, a frequent source of wireless degradation is caused by uncoordinated concurrent networks or by radiating appliances that interfere in the utilized frequency range. Such alien sources of disturbance are emulated by cross-traffic from a hidden terminal in our experiments.

A. Setup of Cross-traffic at Intermediate Hop

We examine the robustness of the networking protocols under cross-traffic using a two-hop topology between a content producer (P) and a consumer (C). Cross-traffic is injected towards an intermediate forwarder (F) as illustrated in Fig. 8. We center (C) and (P) at (F) and verify in preceding measurements that both links perform comparably in both directions. By ensuring symmetry, we prevent a measurement bias with respect to antisymmetric sequencing of the different protocols. (CT) is our cross-traffic generator and placed next to (F), so that (F) overhears all transmissions, while (CT) remains hidden for (C) and (P). Hence, CSMA/CA fails for (C) and (P) and we expect an increased packet loss due to collisions for these nodes.

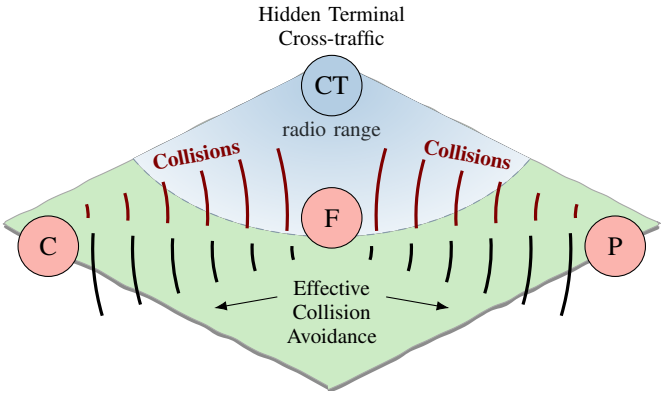


Fig. 8. Experiment setup for measuring protocol resilience under cross-traffic.

To scale the effect of cross-traffic at different stress levels, we configure the traffic generator in two dimensions as illustrated in Fig. 9.

Burst Size reflects the number of consecutive packets sent to a third party link-layer unicast address. Each burst consists of a series of packets with a payload of 100 bytes.

Inter Burst Time denotes the pause in which our cross-traffic generator keeps the radio silent.

With varying cross-traffic patterns in place, we measure the error rates, data load, and time to completion for each protocol. We apply the periodic traffic pattern of one data unit every 1 s advised by our use case. Our first measurement validates

the experimental environment. Fig. 10 displays the times to content arrival in the absence of cross traffic. All protocols perform perfectly as expected.

B. Results

Turning on the cross-traffic generator changes the picture. Fig. 11 presents an overview of the protocol behaviors under 25 different scenarios of competing traffic. The color in each block visualizes the relative packet loss, while the numbers denote the relative redundancy of data packets on the links. A regular, undisturbed data packet traverses each link only once. Numbers higher than 1 indicate duplicate data packets, lower numbers indicate loss on the paths of data or request messages.

Decreasing the pauses between increasing bursts pushes the performance of all protocols below 50 % success rate. Still, the results are quite diverse. While the request-response protocols quickly degrade to error rates above 80 %, those protocols that push data (MQTT-SN and CoAP OBS) show a much higher chance of successfully transmitting data. It should be noted here that the CoAP OBS is unreliable and does not repeat data. Hence, its success rate turns lower than MQTT-SN, while its data rate on the air also drops. On average, only $\approx$ 60 % of the data packets traverse both links, many of which only make the first hop. In contrast, MQTT-SN pushes packets via UDP until an acknowledgment arrives. This leads to a very high redundancy, which almost triples the data rates on the links. By pushing data intensely, though, MQTT-SN manages to attain superior performance among all protocols.

CoAP GET and the ICN protocols transmit data only on request. Since the cross-traffic jamming repeatedly destroys these requests, data is often not even transmitted. In consequence, data only sparsely appears on links even though these reliable protocols retransmit. The results are slightly better for the ICN protocols, since they transfer packets hop-wise with caching in place at the forwarding node.

This harsh, highly disruptive experimental regime reveals a significant heterogeneity among the protocols and their ability to co-exist on stressed links. While MQTT-SN accesses wireless resources rather aggressively – possibly on the expense of concurrent communication, the request-response protocols politely retreat from flooding data onto the congested link. It is noteworthy that data arrives in about equal shares among the four protocol retransmissions, so that about 25 % reaches the consumer only after 8 s. Reducing the retransmission timeouts would further enhance the link utilization and lower the chances of a successful data transfer.

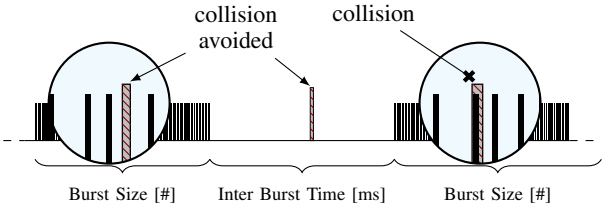


Fig. 9. *Burst size* and *inter burst time* for our generated cross-traffic.

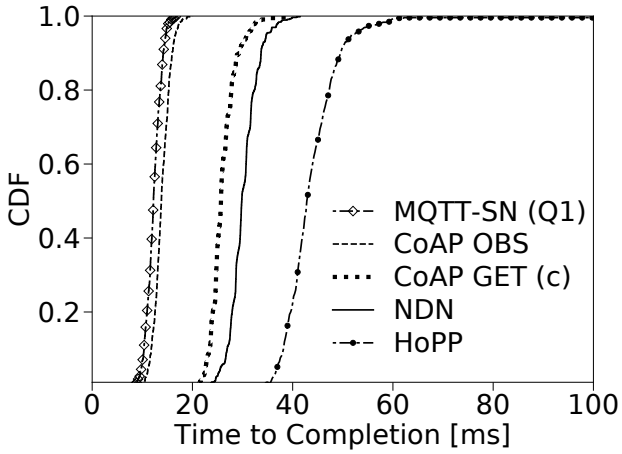


Fig. 10. Time to content arrival in a two-hop topology at 1 s interval without cross-traffic.

VII. Related Work on Protocol Evaluation

A. Data dissemination in the Industrial IoT

Wireless communication plays an important role for connecting sensors and actuators in the IIoT and its heterogeneous systems. We have discussed current wireless link layers in Section III, which are all error prone in the often harsh industrial deployments. Networking protocols on its upper layers may procure for high reliability as well as security needed for application scenarios such as control loops or safety related alerting. Challenges and requirements for typical IIoT scenarios have been investigated in [42]. Bernieri *et al.* [43] monitor factory automation systems and identify traffic anomalies in a hybrid system of traditional Modbus/TCP [44] as well as CoAP communication. Experimental evaluations of a distributed IoT data plane were recently presented in [45] and [46]. While the first work aims at optimizing the overall network throughout on edge nodes, the second introduces a lightweight messaging middleware to minimize resource consumption on low-end devices for edge computing.

Eggert [47] demonstrates on real IoT hardware the feasibility of using QUIC [48] for constrained devices. As a transport based on UDP, it provides a lightweight replacement for TCP with flow-controlled and multiplexed streams, a low-latency connection establishment, and built-in security features, which are valuable additions for safety-critical infrastructures. Extensions, such as Multipath-QUIC [49] and QUIC-FEC [50], bring an improved resiliency to connectivity failures. While comparative evaluations [51], [52] were mainly conducted for general purpose hardware, they yield promising results for deployments using multiple interfaces with high loss probabilities. Multiple interfaces on the same network hierarchy, however, are uncommon in industrial IoT deployments.

B. Performance evaluation of CoAP and MQTT

The performances of CoAP and MQTT have been studied from several perspectives over the last years [53]–[56]. Very early work analyzed the interoperability of specific CoAP

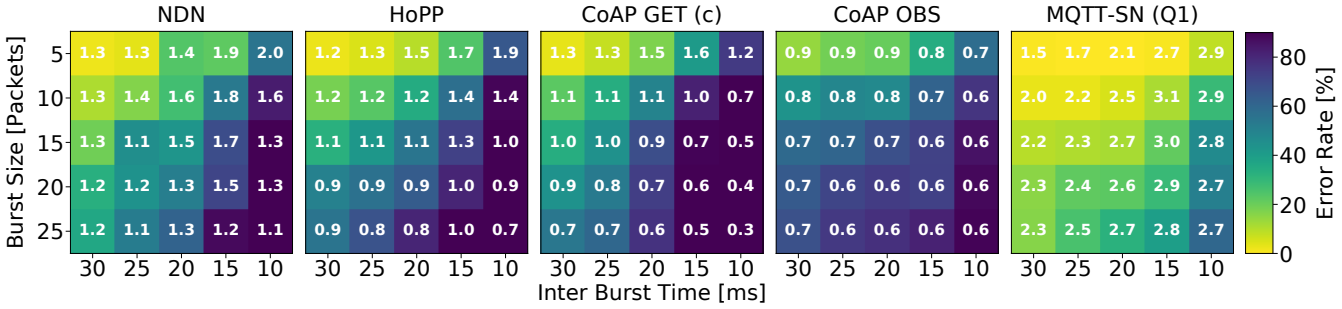


Fig. 11. Error rate vs. data redundancy for a 1 s publishing interval. Colors encode errors and numbers tell the effective ratio of data packets sent over uniquely published items.

implementations [57], [58] without performance evaluation. Later, CoAP implementations have been assessed in comparison to HTTP [59] or on different hardware architectures [60]. MQTT was evaluated in [61] and compared to HTTP in [62]. Rodríguez *et al.* [63] analyze MQTT and HTTP using TCP/IPv4 as a transport. Thangavel *et al.* [64] proposed a common middleware to abstract from CoAP and MQTT. Based on this middleware, CoAP and MQTT were evaluated in a single-hop wired setup. In emulation, MQTT and CoAP have been studied in the context of medical application scenario [65]. Experimental analyses of MQTT and CoAP running on a hardware simulator (Cooja) have been presented by Martí *et al.* [66], and Proos *et al.* [67] perform measurements on a Raspberry Pi. The authors in [68] evaluate implications of the radio technology on higher layers protocols. They focus on the cellular 4G technology Narrowband IoT (NB-IoT). Still, a holistic analysis of these protocols in a consistent experimental setting including many real low-end devices with low-power wireless short range radio technologies is missing.

C. ICN and the IoT

The benefits of ICN/NDN in the IoT have been analyzed mainly from three angles. (*i*) design aspects [69], [70], (*ii*) architecture work [8], [71], [72], and (*iii*) use cases [73]–[75]. By stacking CoAP on ICN, Islam *et al.* [71] introduced CoAP as a convergence layer for applications that can run over both networking worlds. Another approach [76] constructs a pure CoAP deployment option that replicates information-centric properties to gain the beneficial effects of ICN and still sustain protocol compliance with the CoAP specification [6]. Experimental evaluations are supported by several implementations that have become publicly available, including CCN-lite [77] on RIOT [37] and on Contiki [78], and NDN on RIOT [79].

The evaluation of NDN protocol properties in the wild includes the exploitation of NDN communication patterns to improve wireless resource management [80] as well as data delivery on the network layer [81], [82], which are to a larger extent reproducible with data-centric CoAP deployments [76].

We performed a first comparison with common IoT network stacks in [11]. This paper extends our previous work and deepens the analyses in the context of the IIoT.

VIII. Conclusions and Outlook

This paper discussed and analyzed current networking solutions for the constrained Industrial Internet of Things. Starting from the challenging use case of safety-critical sensors and industrial control systems, we derived key requirements for the protocol behavior in a target deployment. Facing these requirements, we deployed and evaluated the three protocol families MQTT-SN, CoAP, and ICN in real-world experiments with settings characteristic for the IIoT.

Our analysis revealed that the choice of protocol largely impacts the application performance. On the overall, lean and simple publish-subscribe protocols such as MQTT-SN and CoAP Observe are versatile and operate efficiently in relaxed environments with low error rates. Request-response schemes hardly meet latency constraints of unscheduled alerts. Even though reliable, MQTT-SN and CoAP quickly fail in massive multi-hop scenarios, in which NDN and NDN-HoPP can both enfold strength of hop-wise transfer and reliably deliver data without the need for significant retransmission rates. MQTT-SN best withstands degradation from cross-traffic of coexisting wireless users—at the price of straining the overall resources by bursty (re-)transmissions.

With these results, we hope to shed light on the role of networking and to strengthen deployment in the constrained IIoT. Our future work will concentrate on progressing distributed IoT applications—facilitated by a robust and versatile Data-centric Web of Things.

Acknowledgment. This work was supported in part by the German Federal Ministry for Education and Research (BMBF) within the projects *I3: Information Centric Networking for the Industrial Internet* and *PIVOT: Privacy-Integrated design and Validation in the constrained IoT*.

A Note on Reproducibility. We explicitly support reproducible research [83], [84]. Our experiments have been conducted in an open testbed. The source code of our implementations (including scripts to set up the experiments, RIOT measurement apps etc.) will be available on Github at https://github.com/5G-I3/Impact-Industrial-IoT-2020.

References

[1] A. Banks and R. G. (Eds.), "MQTT Version 3.1.1," OASIS, OASIS Standard, October 2014. [Online]. Available: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html

[2] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, "5G : A Tutorial Overview of Standards, Trials, Challenges, Deployment, and Practice," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1201–1221, 06 2017.

[3] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks," IETF, RFC 4944, September 2007.

[4] IEEE 802.15 Working Group, "IEEE Standard for Low-Rate Wireless Networks," IEEE, New York, NY, USA, Tech. Rep. IEEE Std 802.15.4™–2015 (Revision of IEEE Std 802.15.4-2011), 2016.

[5] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," IETF, RFC 6550, March 2012.

[6] Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP)," IETF, RFC 7252, June 2014.

[7] E. Rescorla and N. Modadugu, "Datagram Transport Layer Security Version 1.2," IETF, RFC 6347, January 2012.

[8] E. M. Schooler, D. Zage, J. Sedayao, H. Moustafa, A. Brown, and M. Ambrosin, "An Architectural Vision for a Data-Centric IoT: Rethinking Things, Trust and Clouds," in *IEEE 37th Intern. Conference on Distributed Computing Systems (ICDCS)*. Piscataway, NJ, USA: IEEE, June 2017, pp. 1717–1728.

[9] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A Survey of Information-Centric Networking," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, July 2012.

[10] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, k. claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, "Named Data Networking," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 66–73, 2014.

[11] C. Gündogan, P. Kietzmann, M. Lenders, H. Petersen, T. C. Schmidt, and M. Wählisch, "NDN, CoAP, and MQTT: A Comparative Measurement Study in the IoT," in *Proc. of 5th ACM Conference on Information-Centric Networking (ICN)*. New York, NY, USA: ACM, September 2018, pp. 159–171. [Online]. Available: https://doi.org/10.1145/3267955.3267967

[12] J. Chen, X. Cao, P. Cheng, Y. Xiao, and Y. Sun, "Distributed Collaborative Control for Industrial Automation With Wireless Sensor and Actuator Networks," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 12, pp. 4219–4230, 2010.

[13] Q. Wang and J. Jiang, "Comparative Examination on Architecture and Protocol of Industrial Wireless Sensor Network Standards," *IEEE Communications Surveys Tutorials*, vol. 18, no. 3, pp. 2197–2219, 2016.

[14] International Society of Automation, "Wireless Systems for Industrial Automation: Process Control and Related Applications," ISA, Tech. Rep. Standard ISA-100.11a-2011, 2011.

[15] L. M. Feeney, M. Frey, V. Fodor, and M. Günes, "Modes of internetwork interaction in beacon-enabled IEEE 802.15.4 networks," in *14th Mediterranean Ad Hoc Networking Workshop, MED-HOC-NET*. IEEE, June 2015, pp. 1–8.

[16] S. B. Yaala, F. Théoleyre, and R. Bouallegue, "Cooperative resynchronization to improve the reliability of colocated IEEE 802.15.4 -TSCH networks in dense deployments," *Ad Hoc Networks*, vol. 64, pp. 112 – 126, 2017.

[17] A. A. Kumar S., K. Ovsthus, and L. M. Kristensen., "An Industrial Perspective on Wireless Sensor Networks - A Survey of Requirements, Protocols, and Challenges," *IEEE Communications Surveys Tutorials*, vol. 16, no. 3, pp. 1391–1412, 2014.

[18] N. Sastry and D. Wagner, "Security Considerations for IEEE 802.15.4 Networks," in *Proc. of the 3rd ACM Workshop on Wireless Security (WiSe '04)*. New York, NY, USA: ACM, 2004, pp. 32–42.

[19] S. M. Sajjad and M. Yousaf, "Security analysis of IEEE 802.15.4 MAC in the context of Internet of Things (IoT)," in *Conference on Information Assurance and Cyber Security (CIACS '14)*. IEEE, 2014, pp. 9–14.

[20] 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," IETF, RFC 7554, May 2015.

[21] X. Vilajosana, K. Pister, and T. Watteyne, "Minimal IPv6 over the TSCH Mode of IEEE 802.15.4e (6TiSCH) Configuration," IETF, RFC 8180, May 2017.

[22] L. M. Feeney and P. Gunningberg, "Avoiding an IoT 'Tragedy of the Commons'," in *Proc. of the 16th International Conference on Mobile Systems, Applications, and Services (MobiSys '18)*. New York, NY, USA: ACM, 2018, pp. 495–497.

[23] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5g: physical and MAC-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, 2016.

[24] Y. Cai, Z. Qin, F. Cui, G. Y. Li, and J. A. McCann, "Modulation and Multiple Access for 5G Networks," *IEEE Communications Surveys Tutorials*, vol. 20, no. 1, pp. 629–646, 2018.

[25] K. Hartke, "Observing Resources in the Constrained Application Protocol (CoAP)," IETF, RFC 7641, September 2015.

[26] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," IETF, RFC 8446, August 2018.

[27] G. Selander, J. Mattsson, F. Palombini, and L. Seitz, "Object Security for Constrained RESTful Environments (OSCORE)," IETF, RFC 8613, July 2019.

[28] A. Stanford-Clark and H. L. Truong, "MQTT For Sensor Networks (MQTT-SN) Version 1.2," IBM, Protocol Specification, November 2013. [Online]. Available: http://mqtt.org/new/wp-content/uploads/2009/06/MQTT-SN_spec_v1.2.pdf

[29] J. Chen, M. Arumaithurai, L. Jiao, X. Fu, and K. Ramakrishnan, "COPSS: An Efficient Content Oriented Publish/Subscribe System," in *ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS'11)*. Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2011, pp. 99–110.

[30] M. Zhang, V. Lehman, and L. Wang, "Scalable name-based data synchronization for named data networking," in *IEEE INFOCOM'17*, ser. INFOCOM '17. Los Alamitos, CA, USA: IEEE Computer Society, 2017, pp. 1–9.

[31] C. Gündogan, P. Kietzmann, T. C. Schmidt, and M. Wählisch, "HoPP: Robust and Resilient Publish-Subscribe for an Information-Centric Internet of Things," in *Proc. of the 43rd IEEE Conference on Local Computer Networks (LCN)*. Piscataway, NJ, USA: IEEE Press, Oct. 2018, pp. 331–334. [Online]. Available: http://doi.org/10.1109/LCN.2018.8638030

[32] T. C. Schmidt, S. Wölke, N. Berg, and M. Wählisch, "Let's Collect Names: How PANINI Limits FIB Tables in Name Based Routing," in *Proc. of 15th IFIP Networking Conference*. Piscataway, NJ, USA: IEEE Press, May 2016, pp. 458–466.

[33] J. H. Saltzer, D. P. Reed, and D. D. Clark, "End-to-End Arguments in System Design," *ACM Trans. Comput. Syst.*, vol. 2, no. 4, pp. 277–288, Nov 1984.

[34] X. de Carnavalet and M. Mannan, "Killed by Proxy: Analyzing Client-end TLS Interception Software," in *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2016.

[35] R. Holz, T. Riedmaier, N. Kammenhuber, and G. Carle, "X.509 Forensics: Detecting and Localising the SSL/TLS Men-in-the-Middle," in *Computer Security – ESORICS 2012*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 217–234.

[36] C. Gündogan, C. Amsüss, T. C. Schmidt, and M. Wählisch, "IoT Content Object Security with OSCORE and NDN: A First Experimental Comparison," in *Proc. of 19th IFIP Networking Conference*. Piscataway, NJ, USA: IEEE Press, June 2020, pp. 19–27. [Online]. Available: https://ieeexplore.ieee.org/document/9142731

[37] E. Baccelli, C. Gündogan, O. Hahm, P. Kietzmann, M. Lenders, H. Petersen, K. Schleiser, T. C. Schmidt, and M. Wählisch, "RIOT: an Open Source Operating System for Low-end Embedded Devices in the IoT," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4428–4440, December 2018. [Online]. Available: http://dx.doi.org/10.1109/JIOT.2018.2815038

[38] C. Bormann, M. Ersue, and A. Keranen, "Terminology for Constrained-Node Networks," IETF, RFC 7228, May 2014.

[39] C. Bormann, "6LoWPAN-GHC: Generic Header Compression for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)," IETF, RFC 7400, November 2014.

[40] C. Gündogan, T. C. Schmidt, M. Wählisch, C. Scherb, C. Marxer, and C. Tschudin, "ICN Adaptation to LowPAN Networks (ICN LoWPAN)," IRTF, IRTF Internet Draft – work in progress 10, February 2021. [Online]. Available: https://tools.ietf.org/html/draft-irtf-icnrg-icnlowpan

[41] Atmel, *Low Power 2.4 GHz Transceiver for ZigBee, IEEE 802.15.4, 6LoWPAN, RF4CE, SP100, WirelessHART, and ISM Applications*, Atmel Corporation, September 2009. [Online]. Available: http://www.atmel.com/images/doc8111.pdf

[42] G. Hansch, P. Schneider, and G. S. Brost, "Deriving Impact-Driven Security Requirements and Monitoring Measures for Industrial IoT," in *Proc. of the 5th on Cyber-Physical System Security Workshop (CPSS'19)*. New York: ACM, 2019, pp. 37–45.

[43] G. Bernieri, M. Conti, and G. Pozzan, "AMON: An Automaton MONitor for Industrial Cyber-Physical Security," in *Proc. of the 14th International Conference on Availability, Reliability and Security (ARES '19)*. New York: ACM, 2019, pp. 1–10.

[44] Modbus–IDA, "Modbus application protocol specification v1. 1b," Modbus–IDA, Tech. Rep., 2006.

[45] M. Nolan, M. J. McGrath, M. Spoczynski, and D. Healy, "Adaptive Industrial IOT/CPS Messaging Strategies for Improved Edge Compute Utility," in *Proc. of the Workshop on Fog Computing and the IoT (IoT-Fog '19)*. New York: ACM, 2019, pp. 16–20.

[46] B. Chun, B. Oh, C. Cho, and D. Lee, "Design and Implementation of Lightweight Messaging Middleware for Edge Computing," in *Proceedings of the 6th International Conference on Control, Mechatronics and Automation (ICCMA '18)*. New York: ACM, 2018, pp. 170–174.

[47] L. Eggert, "Towards Securing the Internet of Things with QUIC," in *Proc. of 3rd NDSS Workshop on Decentralized IoT Systems and Security (DISS)*. Internet Society (ISOC), 2020.

[48] J. Iyengar and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport," IETF, Internet-Draft – work in progress 34, January 2021.

[49] Q. D. Coninck and O. Bonaventure, "Multipath Extensions for QUIC (MP-QUIC)," IETF, Internet-Draft – work in progress 07, May 2021.

[50] I. Swett, M.-J. Montpetit, V. Roca, and F. Michel, "Coding for QUIC," IETF, Internet-Draft – work in progress 04, March 2020.

[51] Q. D. Coninck and O. Bonaventure, "Multipath QUIC: Design and Evaluation," in *Proc. of CoNEXT '17*. New York, NY, USA: ACM, Dec. 2017, pp. 160–166.

[52] F. Michel, Q. D. Coninck, and O. Bonaventure, "QUIC-FEC: Bringing the benefits of Forward Erasure Correction to QUIC," in *Proc. of 19th IFIP Networking Conference*. Piscataway, NJ, USA: IEEE Press, May 2019, pp. 1–9.

[53] M. Iglesias-Urkia, A. Orive, and A. Urbieta, "Analysis of CoAP Implementations for Industrial Internet of Things: A Survey," *Procedia Computer Science*, vol. 109, pp. 188–195, 2017.

[54] J. Dizdarevic, F. Carpio, A. Jukan, and X. Masip-Bruin, "Survey of Communication Protocols for Internet-of-Things and Related Challenges of Fog and Cloud Computing Integration," *ACM Comput. Surv.*, vol. 51, no. 6, pp. 116–1 – 116–29, Jan. 2019.

[55] D. Koutras, G. Stergiopoulos, T. Dasaklis, P. Kotzanikolaou, D. Glynos, and C. Douligeris, "Security in IoMT Communications: A Survey," *Sensors*, vol. 20, no. 17, p. 4828, 2020.

[56] N. F. Syed, Z. Baig, A. Ibrahim, and C. Valli, "Denial of service attack detection through machine learning for the IoT," *Journal of Information and Telecommunication*, vol. 4, no. 4, pp. 482–503, 2020.

[57] C. Lerche, K. Hartke, and M. Kovatsch, "Industry adoption of the Internet of Things: A constrained application protocol survey," in *Proc. 17th IEEE International Conf on Emerging Technologies & Factory Automation (ETFA)*. Piscataway, NJ, USA: IEEE, 2012, pp. 1–6.

[58] B. C. Villaverde, D. Pesch, R. de Paz Alberola, S. Fedor, and M. Boubekeur, "Constrained Application Protocol for Low Power Embedded Networks: A Survey," in *Proc. of 6th International Conf on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*. Washington, DC, USA: IEEE Computer Society, 2012, pp. 702–707.

[59] A. Ludovici, P. Moreno, and A. Calveras, "TinyCoAP: A Novel Constrained Application Protocol (CoAP) Implementation for Embedding RESTful Web Services in Wireless Sensor Networks Based on TinyOS," *J. Sensor and Actuator Networks*, vol. 2, no. 2, pp. 288–315, 2013.

[60] C. P. Kruger and G. P. Hancke, "Benchmarking Internet of things devices," in *Proc. of 12th IEEE International Conf on Industrial Informatics (INDIN)*. Piscataway, NJ, USA: IEEE, 2014, pp. 611–616.

[61] A. Elmangoush, R. Steinke, T. Magedanz, A. A. Corici, A. Bourreau, and A. Al-Hezmi, "Application-derived communication protocol selection in M2M platforms for smart cities," in *Proc. of 18th International Conference on Intelligence in Next Generation Networks (ICIN)*. Piscataway, NJ, USA: IEEE, 2015, pp. 76–82.

[62] D. Mishra, R. S. Yadav, K. K. Agrawal, and A. Abbas, "Study of Application Layer Protocol for Real-Time Monitoring and Maneuvering," in *International Conference on Innovative Computing and Communications*, A. Khanna, D. Gupta, S. Bhattacharyya, V. Snasel, J. Platos, and A. E. Hassanien, Eds. Singapore: Springer Singapore, 2020, pp. 439–449.

[63] J. J. R. Rodríguez, J. F. C. García, and E. J. A. üello Prada, "Toward Automatic and Remote Monitoring of the Pain Experience: An Internet of Things (IoT) Approach," in *Applied Technologies*, M. Botto-Tobar, M. Z. Vizuete, P. Torres-Carrión, S. M. León, G. P. Vásquez, and B. Durakovic, Eds. Cham: Springer International Publishing, 2020, pp. 194–206.

[64] D. Thangavel, X. Ma, A. Valera, H.-X. Tan, and C. K.-Y. Tan, "Performance evaluation of MQTT and CoAP via a common middleware," in *Proc. of ISSNIP*. Piscataway, NJ, USA: IEEE, 2014, pp. 1–6.

[65] Y. Chen and T. Kunz, "Performance evaluation of IoT protocols under a constrained wireless access network," in *International Conference on Selected Topics in Mobile & Wireless Networking (MoWNeT)*. Piscataway, NJ, USA: IEEE, 2016, pp. 1–7.

[66] M. Martí, C. Garcia-Rubio, and C. Campo, "Performance Evaluation of CoAP and MQTT-SN in an IoT Environment," in *13th Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI'19)*, vol. 31, 2019, p. 49.

[67] D. P. Proos and N. Carlsson, "Performance Comparison of Messaging Protocols and Serialization Formats for Digital Twins in IoV," in *Proc. of 19th IFIP Networking Conference*. Piscataway, NJ, USA: IEEE Press, June 2020, pp. 10–18.

[68] A. Larmo, A. Ratilainen, and J. Saarinen, "Impact of CoAP and MQTT on NB-IoT system performance," *Sensors*, vol. 19, p. 7, 2018.

[69] W. Shang, Y. Yu, T. Liang, B. Zhang, , and L. Zhang, "NDN-ACE: Access Control for Constrained Environments over Named Data Networking," NDN, Technical Report NDN-0036, December 2015.

[70] B. Mathieu, C. Westphal, and P. Truong, "Towards the usage of ccn for iot networks," in *Internet of Things (IoT) in 5G Mobile Technologies*. Cham, Switzerland: Springer, 2016, pp. 3–24.

[71] H. M. A. Islam, D. Lagutin, A. Ylä-Jääski, N. Fotiou, and A. V. Gurtov, "Transparent CoAP Services to IoT Endpoints through ICN Operator Networks," *Sensors*, vol. 19, no. 6, p. 1339, 2019.

[72] A. L. R. Madureira, F. R. C. Araújo, G. B. Araújo, and L. N. Sampaio, "NDN Fabric: Where the Software-Defined Networking Meets the Content-Centric Model," *IEEE Transactions on Network and Service Management*, 2020.

[73] J. Burke, P. Gasti, N. Nathan, and G. Tsudik, "Securing Instrumented Environments over Content-Centric Networking: the Case of Lighting Control and NDN," in *Computer Communications Workshops (INFOCOM WKSHPS), 2013 IEEE Conference on*. Piscataway, NJ, USA: IEEE, 2013, pp. 394–398.

[74] M. Amadeo, C. Campolo, A. Iera, and A. Molinaro, "Information Centric Networking in IoT scenarios: The case of a smart home," in *Proc. of IEEE International Conference on Communications (ICC)*. Piscataway, NJ, USA: IEEE, June 2015, pp. 648–653.

[75] M. Frey, C. Gündogan, P. Kietzmann, M. Lenders, H. Petersen, T. C. Schmidt, F. Shzu-Juraschek, and M. Wählisch, "Security for the Industrial IoT: The Case for Information-Centric Networking," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT) (WF-IoT 2019)*. Piscataway, NJ, USA: IEEE Press, April 2019, pp. 424–429. [Online]. Available: http://dx.doi.org/10.1109/WF-IoT.2019.8767183

[76] C. Gündogan, C. Amsüss, T. C. Schmidt, and M. Wählisch, "Toward a RESTful Information-Centric Web of Things: A Deeper Look at Data Orientation in CoAP," in *Proc. of 7th ACM Conference on Information-Centric Networking (ICN)*. New York: ACM, September 2020, pp. 77–88. [Online]. Available: https://doi.org/10.1145/3405656.3418718

[77] C. Tschudin, C. Scherb *et al.*, "CCN Lite: Lightweight implementation of the Content Centric Networking protocol," 2018. [Online]. Available: http://ccn-lite.net

[78] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki - A Lightweight and Flexible Operating System for Tiny Networked Sensors." in *Proc. of IEEE Local Computer Networks (LCN)*. Los Alamitos, CA, USA: IEEE Computer Society, 2004, pp. 455–462.

[79] W. Shang, A. Afanasyev, and L. Zhang, "The Design and Implementation of the NDN Protocol Stack for RIOT-OS," in *Proc. of IEEE GLOBECOM 2016*. Washington, DC, USA: IEEE, 2016, pp. 1–6.

[80] P. Kietzmann, C. Gündogan, T. C. Schmidt, O. Hahm, and M. Wählisch, "The Need for a Name to MAC Address Mapping in NDN: Towards Quantifying the Resource Gain," in *Proc. of 4th ACM Conference on Information-Centric Networking (ICN)*. New York, NY, USA: ACM, September 2017, pp. 36–42.

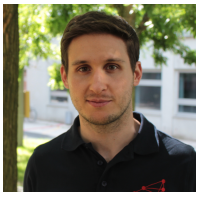
[81] E. Baccelli, C. Mehlis, O. Hahm, T. C. Schmidt, and M. Wählisch, "Information Centric Networking in the IoT: Experiments with NDN in the Wild," in *Proc. of 1st ACM Conf. on Information-Centric Networking (ICN-2014)*. New York: ACM, September 2014, pp. 77–86. [Online]. Available: http://dx.doi.org/10.1145/2660129.2660144

[82] C. Gündogan, J. Pfender, P. Kietzmann, T. C. Schmidt, and M. Wählisch, "On the Impact of QoS Management in an Information-centric Internet of Things," *Computer Communications*, vol. 154, pp. 160–172, March 2020. [Online]. Available: https://doi.org/10.1016/j.comcom.2020.02.046

[83] Q. Scheitle, M. Wählisch, O. Gasser, T. C. Schmidt, and G. Carle, "Towards an Ecosystem for Reproducible Research in Computer Networking," in *Proc. of ACM SIGCOMM Reproducibility Workshop*. New York, NY, USA: ACM, August 2017, pp. 5–8.

[84] ACM, "Result and Artifact Review and Badging," http://acm.org/publications/policies/artifact-review-badging, Jan., 2017.

AUTHOR BIOGRAPHY



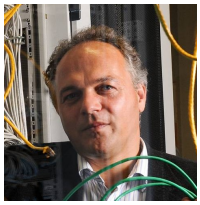
Cenk Gündoğan received the M.Sc. degree in computer science from the Institut für Informatik, Freie Universität Berlin, Germany, in 2016. Currently, he is pursuing the Ph.D. degree with the Internet Technologies Group, Hamburg University of Applied Sciences, Germany, and explored within the I3 project—Information Centric Networking (ICN) for the Industrial Internet—routing, QoS, and resilience in ICN-based and IoT tailored networks. Recently, Cenk Gündoğan put focus on a data-centric Web of Things deployment option by applying ICN principles to the IETF envisioned IoT network stack. He is one of the core developers and maintainer of RIOT.

Peter Kietzmann received the M.Eng. degree in information technology from the Hamburg University of Applied Sciences, Hamburg, Germany, where he is currently pursuing the Ph.D. degree with the Internet Technologies Research Group. His particular research interest includes low-power radios, and IoT protocols, many of which he analyzed and transformed into code of RIOT. In the German research project I3 (ICN for the Industrial Internet) he explores IoT-based technologies for information centric networks and security components.

Martine S. Lenders is a researcher at the of the Internet Technologies research group at Freie Universität Berlin. She obtained both her B.Sc. (2011) and M.Sc. (2016) in Computer Science at Freie Universität Berlin with focus on the development and comparison of several IP-based network stacks for embedded devices. Her research interests include the Internet of Things, Information Centric Networking, and API design. Accompanying her work as researcher she is currently pursuing her PhD in those fields. Martine Sophie Lenders is also one of core developers of the operating system RIOT and maintains large parts of its networking infrastructure.

Hauke Petersen is a research associate at Freie Universität Berlin, where he is pursuing his Ph.D. in the fields of operating systems and networking for constrained devices in the Internet of Things. After obtaining a B.Sc. in Computer Science from Hochschule Darmstadt and a M.Sc. in Computer Engineering from the Technical University of Denmark, he worked as a Systems and Safety engineer for the German automotive industry, before joining Freie Universität Berlin. Particularly passionate about combining real-time control systems with Internet connectivity and implementing this on very constrained hardware platforms, Hauke Petersen is one of the core developers and maintainers of RIOT OS, and published articles on such topics in high-quality venues including ACM and IEEE conferences.

Michael Frey received his M.Sc. in Computer Science at RheinMain University of Applied Sciences, Wiesbaden, Germany in 2010. He is senior software engineer at Safety IO where he focuses on tackling challenges bringing ICN to safety-critical environments. His research interests include routing and network management in ICN-based IIoT networks.



Thomas C. Schmidt is professor of Computer Networks and Internet Technologies at Hamburg University of Applied Sciences (HAW), where he heads the Internet Technologies research group (iNET). Prior to moving to Hamburg, he was director of a scientific computer centre in Berlin. He studied mathematics, physics and German literature at Freie Universitaet Berlin and University of Maryland, and received his Ph.D. from FU Berlin in 1993. Since then he has continuously conducted numerous national and international research projects. He was the principal investigator in a number of EU, nationally funded and industrial projects as well as visiting professor at the University of Reading, U.K.. His continued interests lie in the development, measurement, and analysis of large-scale distributed systems like the Internet. He serves as co-editor and technical expert in many occasions and is actively involved in the work of IETF and IRTF. Together with his group he pioneered work on an information-centric Industrial IoT and the emerging data-centric Web of Things. Thomas is a co-founder of several large open source projects and coordinator of the community developing the RIOT operating system - the friendly OS for the Internet of Things.

Felix Shzu-Juraschek is an IoT system architect for safety products that protect people and facility infrastructures at Safety io, a subsidiary of MSA The Safety Company. Felix heads MSA's global connectivity steering team to accelerate the connection of IoT devices and SaaS applications for safety-critical environments, such as the fire service and other markets. He graduated from Freie Universität Berlin after studying Computer Science and Psychology. He received his Ph.D. in Computer Science from the Humboldt Universität Berlin in 2015.

Matthias Wählisch is an Assistant Professor of Computer Science at Freie Universität Berlin where he heads the Internet Technologies Research Lab. He received his Ph.D. in computer science with highest honors from Freie Universität Berlin. His research and teaching focus on efficient, reliable, and secure Internet communication. This includes the design and evaluation of networking protocols and architectures, as well as Internet measurements and analysis. His efforts are driven by improving Internet communication based on sound research.. Matthias is the PI of several national and international projects, supported by overall 4.7M EUR grant money. He published more than 150 peer-reviewed papers (*e.g.,* at ACM HotNets, ACM IMC, The Web Conference). Since 2005, Matthias is active within IETF/IRTF, including eight RFCs and several Internet drafts. His research results have been distinguished multiple times. Amongst others, he received the Young Talents Award of Leibniz-Kolleg Potsdam for outstanding achievements in advancing the Internet, as well as the Excellent Young Scientists Award (10,000 EUR) for his contributions to the Internet of Things and their prospective entrepreneurial practice. He co-founded some successful open source projects such as RIOT, where he is still responsible for the strategic development.