Cloud-based WirelessHART networking for Critical Industrial Monitoring and Control

Leonardo Ascorti[1], Stefano Savazzi[2], Stefano Galimberti[1]
[1]Pepperl-Fuchs, Sulbiate (MI), Italy [2]National Research Council of Italy CNR-IEIIT, Milano Italy

***Abstract*— Cloud-enabled wireless industrial sensor networks is an emerging paradigm in machine-type communication. It supports "device-centric" architectures that efficiently exploit intelligence at field device side replacing Host-centric architectures to handle critical condition. In this paper we propose a flexible architecture for integrating a self-contained network-embedded cloud system (Wireless Cloud Network, WCN) with a wireless industrial network infrastructure implementing the Time Synchronized Channel Hopping (TSCH) as supported by commercial WirelessHART systems. The WCN acts as a small-cell embedded network consisting of devices that gather and process pervasive information about the state of the industrial plant. Devices member of the cloud support advanced communication services and enable early and localized detection of dangerous conditions. In addition, they act as data processing centers distributed at the edge of the TSCH network. An hardware and software architecture is developed and tailored for WirelessHART (IEC 62591) protocol while preliminary experimental measurements are also carried out to evaluate the feasibility and the effectiveness of the proposed system.**

I. INTRODUCTION

The adoption of wireless communication and sensor networks in the industrial context is becoming of strategic interest for manufacturers and plant designers [1]. Next generation wireless sensor network technologies are expected to be integrated into the Internet of Things (IoT) paradigm [2], allowing for the global interconnection of heterogeneous smart physical objects with advanced functionalities. Following the paradigm of IoT, emerging technologies inside modern factories are evolving to intelligent environments, where wireless network technologies and mobile information access are now playing a key role for the efficient design of industrial processes.

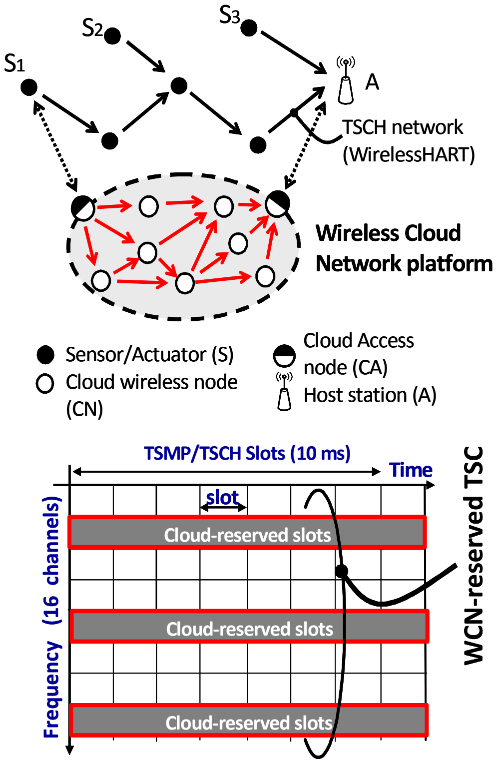


Fig. 1. Wireless Cloud Network (WCN) platform over TSCH network (WirelessHART).

Although many technological solutions and standards (e.g., WiFi/LTE, IEC standards [3]) have been investigated for application-specific contexts, even if the proposed solutions effectively address consumer needs, they are not yet fully ready for industrial applications with high safety, reliability, security and real-time requirements [1]. It is generally acknowledged that to allow for a wider adoption of wireless networks in an industrial context some substantial technology innovation is required. In particular in terms of: (i) novel system architectures, e.g., the introduction of new types of nodes or new functions in existing ones; (ii) self-configuring and learning protocols; (iii) advanced communication strategies to support delay/safety-critical applications. Similarly as nowadays evolution of cellular networks towards 5G technologies [4] where designs have historically relied on the role of ‘cells’ as fundamental units within the radio access network, it is expected that current “Host-centric” architecture concept of commercial wireless industrial systems may change. Over the last few years, different trends have been pointing to a disruption of host-centric structure in favor of “device-centric” architectures where the design approach is to exploit also intelligence at the sensor/field device side and to allow communication among peer nodes based on advanced device-

---

This work has been performed in the framework of the EU research project DIWINE - Dense Cooperative Wireless Cloud Network - under FP7 ICT Objective 1.1 - The Network of the Future.

to-device (D2D) radio access technologies (RAT).

Cloud-enabled wireless industrial sensor networks [5] is an emerging paradigm in machine-type communication: it consists of self-organizing, massively dense air-interacting wireless devices (serving as sensors, actuators, relays or distributed computing nodes) that have the ability to gather and process pervasive information about the state of the industrial plant, by enabling early detection and localization of any dangerous situation. This paper proposes an extensible and flexible architecture for integrating an (embedded) wireless cloud network system (WCN) with a wireless industrial sensor network. As illustrated in Fig. 1, the cloud system consists of cloud devices and nodes serving as distributed access points for field devices an wireless instrumentation outside the WCN system. Cloud devices mutually interconnect among each other and provide advanced cloud services to the TSCH network (consisting of Host and I/O sensors connected in mesh mode). The cloud network is capable of hosting distributed applications and services and it is designed to efficiently interact with a Time Synchronized Channel Hopping (TSCH) protocol implementation [6] (i.e., WirelessHART, standard IEC 62591).

A. Contributions

During the last few years, many researchers have investigated on ways to interconnect wireless sensor networks to cloud-based infrastructures [7]. Much of the previous work has been focused on theoretical aspects of system architecture rather than actual deployment and testing. The proposed cloud system is tailored for industrial settings as it acts as a self-contained embedded network [8] consisting special field devices or "cloud devices" that mutually interconnect among each other (thus mimicking a small-cell system), and act as data processing centers distributed at the edge of the TSCH network. The proposed solution thus entails the loosely coupled coexistence, at the level of each wireless device, supporting any industry-standard wireless protocol stack. According to this device-centric architectural design principle, cloud devices do not simply serve as possibly faulty endpoints but also as critical computing and storage resources that must be reliably connected to satisfy tight quality-of-service (QoS) constraints. The specifics of the architecture are outlined in Sect. 2. Relevant industrial application cases are also identified where the use of the cloud can overcome the limitations of currently deployed TSCH based industry standards wireless systems. Relevant network processing algorithms inside the cloud system are addressed in Sect. 3. In Sect. 4 the proposed hardware and software architecture are tailored for WirelessHART release, while preliminary experimental measurements are carried out to evaluate the feasibility and the effectiveness of the proposed system.

II. CLOUD-BASED TSCH: EXTENDED SERVICE SET

In a Process Automation context most currently installed Wireless Sensor Networks (WSNs) adopt a Time-Synchronized Channel Hopping (TSCH) protocol on top of the IEEE 802.15.4 physical layer (PHY) access scheme. TSCH networks are topology independent and can be used in star topologies as well as partial or full mesh topologies: the approach has been also recently standardized in the new amendment IEEE 802.15.4e [6]. The TSCH architectural solution can be classified as "Host-centric" and it was primarily driven by the requirements of making possible battery-powered sensors implementation with a battery life of many years. For several industrial applications (e.g., supervisory control and monitoring applications with a static – or semi-static – upstream sampling rate), these architecture implementations were in fact able to perform in a largely satisfactory way [3], both from a reliability and latency point of view. On the contrary, due to their optimized design for energy consumption and deterministic traffic management TSCH based solutions provide applications with limited scheduling options, thereby making some workloads required in specific industrial applications (as detailed the following sections) more difficult to handle.

As depicted in Fig. 1, the proposed cloud-based architecture consists of a TSCH compliant industrial network underlaid with a distributed and self-contained network of cloud devices referred to as Wireless Cloud Network (WCN) sharing the same spectrum resources and supporting special PHY/MAC functionalities and applications. Cloud devices adopt TSCH to acquire synchronization with the Host station, while they autonomously self-organize to meet specific service requirements not supported by the host-centric industrial system. In the example messages (i.e., data frames) originated from the TSCH network can be off-loaded through the WCN network consisting of cloud devices (C) and access nodes (CA). Cloud devices (C) are cooperating directly at PHY/MAC creating a powerful "virtual" relay node that supports advanced distributed internal processing. The cloud access nodes (CA) act as distributed cloud controllers and provide a simple PHY/MAC interface to TSCH devices (outside the WCN system) requesting for cloud resources. To guarantee full compatibility with TSCH solution, each cloud node does support both TSCH protocol and the new cloud functions allowing for fully parallel network operation – when so desired. In the following sections we outline a number of extended services that can be supported by the cloud to address relevant industrial applications scenarios. For all cases, it is assumed the use of battery-equipped devices while energy usage should be properly taken into account.

A. Real-time critical process automation services

In this section three real-time and safety critical process automation scenarios are identified which are not properly addressed by current industrial sensor networks implementations based on TSCH technology. For each main scenario, a few more specific Application Cases are then described in detail – along with the associated requirements and expected performance targets of the cloud section. These requirements are taken as a reference so as to focus the theoretical investigations (see Sect. 3) on specific industrial scenarios. The performance goals stated in the following sections assume the constraints related to a battery-powered application scenario where frequent device servicing (e.g. for battery substitution) is not viable and energy efficiency is thus mandatory. In this context, the target is for around an order of magnitude performance improvement with respect to the current industrial state-of-the art.

Delay-critical reaction. This case suggests the use of the cloud network for fast servicing of asynchronously generated and sporadic "events". In typical settings this happens when the Host control system detects some unexpected condition that requires an action over the network in a low-latency mode. This use case is mainly related with a "downstream" (i.e. Host to sensors) transfer of a small amount information with a high level of reliability. Relevant application examples that conform with the selected type are listed in the following:

Level-triggered Control. This case refers to the situation characterized by one or more sensor device variables associated with a hysteretic control strategy based on threshold detection, e.g. in a level measurement context. A 'discrete' actuation action (typically binary) needs to be carried out by the cloud section on one – or more – sensor nodes so that the control strategy is fulfilled with the shortest possible latency.

Dynamic Publishing. A sensor node identifies an abnormal condition which makes desirable a temporary – but immediate – increase of the pre-programmed publishing rate for one – or more – sensor device variables. A substantial, e.g. an order of magnitude, increase of the programmed publishing rate should be made possible by the cloud in a comparatively short time. The publishing-rate increase could be obtained either by additional bandwidth allocation (e.g., by increasing the PHY data rate of devices) and/or network operating modes or topology changes. The maximum delay endured should be 1000 ms, with a desired goal of 500 ms.

High data transfer. This scenario refers to the situation where a somewhat large amount of delay-tolerant data needs to be transferred. The use case might require symmetric "downstream" or "upstream" information transfer, even if these are not expected to take place in parallel. As detailed in the following typical application examples include over-the-air programming of devices deployed on field or acquisition of logging-information.

Over-the-Air Device Software Upgrade. The software – or a data section – within a sensor node needs to be periodically upgraded so that the device can be properly re-configured for a modified application case; the upgrade entails the transfer of a relevant amount of data. A substantial increase of the baseline network downstream throughput should be made possible by the cloud so that the overall software upgrade time to the sensor is minimized.

Waveform Table Transfer. This use case occurs when a sensor node has memorized a relevant amount of data associated with the so called 'waveform table', e.g. as resulting from a frequency-spectrum acquisition performed by a vibration-sensing field device. A substantial increase of the baseline network upstream throughput should be made possible by the cloud so that the overall transfer time of the required data block from the sensor is minimized.

Critical control. This use case focuses on a scenario where closed loop, periodic and real-time control actions need to be performed. In typical settings the Host station sets-up a control strategy connecting one (or more) sensor measurements with actuators to stabilize a process according to user specified set-points (some relevant application cases are primary flow and pressure control). Critical control is related with both a "downstream" and an "upstream" periodic information transfer, where the downstream information (feedback control) is typically more critical.

B. Decentralized monitoring services

In current TSCH implementations process and network state estimation (or monitoring) are performed centrally by the Host station collecting observations from field devices. The WCN architecture can instead support decentralized consensus-based estimation of these parameters. The cloud devices are thus configured to gather and exchange local observations about any physical process state to be monitored or controlled in such a way to allow early detection and more efficient sensing of any dangerous situation.

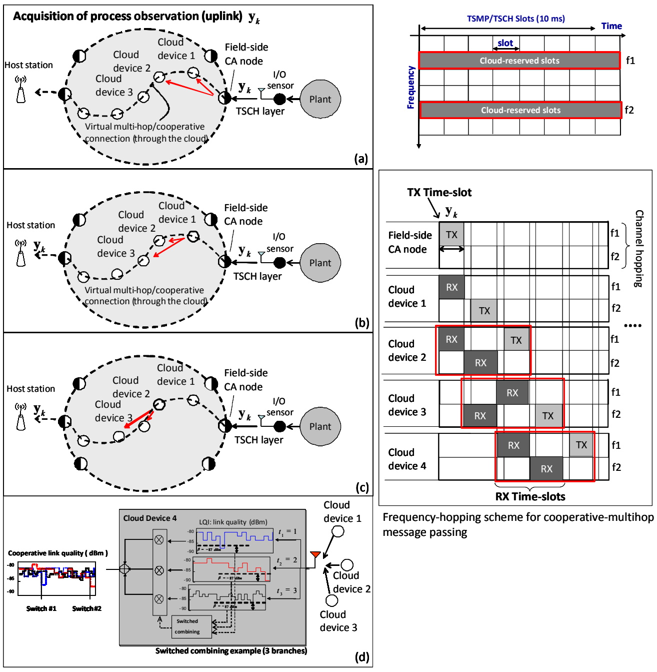
Autonomous diagnostic and learning of the network state. In this case the cloud is tailored to support the autonomous learning of TSCH network state. Each cloud device retrieves critical parameters about the local network status around the device (topology information), the device health (e.g., using the response to HART command 779) and share those information with neighbor devices based on consensus-based processing. The *local* knowledge about the TSCH *global* network status allows each device to perform autonomous diagnostic and comparative performance tests. Application examples are listed in the following.

Localized interference detection. Local measurements acquired from spectrum analysis (i.e., energy detection over multiple PHY channels) can be shared among devices to localize possible interference signals over unlicensed 2.4GHz.

Link Quality Indication estimation. The cloud can support the distributed estimation of key macroscopic propagation parameters that impact on the overall network state [15].

Safety-critical global data sharing for monitoring of process states. Each cloud device is designed to monitor any physical quantity describing an environment (temperature, pressure etc...), share those information with other cloud nodes and finally acquire a global consensus about the state of the considered environment. Critical conditions are identified by comparative analysis between local and global estimation of physical states.

Localized temperature (or pressure) increase. The cloud can be configured to detect any local critical condition by comparative analysis (e.g., temperature higher than neighbor average). Both a direct local action (device-level) or a fast Host indication (i.e., through the cloud) should be considered.



Frequency-hopping scheme for cooperative-multihop message passing

Fig. 2. Cooperative-multihop architecture for WCN. Example of reliable acquisition of process observation: cooperative-multihop frame transmission scheme inside the cloud for redundancy 2 (a-c) and switched combining scheme (d). Synchronous frequency-hopped time-division scheme for message passing along the multi-hop route (right side).

Environment reconstruction and radio localization. Cloud devices can be also designed to allow for the real-time reconstruction of the environment by combining multiple views from local devices [9]. Environmental detection can be implemented for example by monitoring the radio-frequency (RF) field to detect the presence of people or object inside confined critical areas or hazardous work-spaces [10].

III. CLOUD NETWORK PROCESSING

In this section relevant cloud network processing are illustrated and reviewed. Incremental redundancy techniques are first evaluated implementing highly reliable mesh communication based on the use of hybrid cooperative and mesh/multihop networking. Next, the use of cloud is investigated for decentralized estimation (by consensus-based techniques [9]) of critical plant parameters. For all cases the cloud nodes are configured to implement enhanced IEEE 802.15.4 PHY data-rate transmission [11]: the goal is to support high rate applications where explicit device-level acknowledgements are likely not required.

A. Cooperative-multihop processing

Some recent attempts in the literature have been made towards the definition of MAC specifics to enable cooperative communication over IEEE 802.15.4 networks (see for example [12]), although the topic is still considered an open issue. The incorporation of the cooperative network architecture into any wireless industrial system implementation is expected to be particularly effective when reliability of communication is the primary concern [13].By adopting incremental redundancy and implicit acknowledgement, the cooperative techniques are promising for several delay-critical reaction use cases. Let the WCN network in Fig. 2(a) be represented by a set of randomly distributed cloud nodes within a specific area. A sequence of messages is continuously transmitted between a pair of CA nodes acting as *source* node S and *destination* node D over an optimal "connection oriented" unicast route path $\mathcal{R}$ (primary route) that involves M cloud nodes relaying data to destination D. The propagation of consecutive messages along the route path is based on frequency-hopped time division access as illustrated on the right side of Fig. 2. The *cooperative-multihop* architecture herein introduced improves the reliability of multi-hop message passing along the primary route by implementing a chain of consecutive cooperative transmissions, where each transmitted data frame is overheard by d receivers along the multi-hop path. In the example of Fig. 2, the field-side CA interacts with the corresponding I/O sensor (field device) using standard TSCH commands (e.g., WirelessHART) and forwards to the cloud any critical observation of the plant state y_k to be used by the Host station to process control commands. The cloud devices implement a reliable virtual connection between the field and host-side CA node pair.

The proposed cooperative protocol is outlined in Fig. 2(b)-(c). At time slot $t = 1$ (for convenience the time slots are numbered as for the nodes) the process observation originated from I/O sensor is forwarded by the field-side CA acting as source S to cloud devices 1 and 2 (for $d = 2$ in the example). The same data frame is now relayed (at time $t = 2$) from cloud device 1 to devices 2 (and 3) and so on: cloud device 2 can now exploit two copies of the same message received from two different PHY channels and experiencing independent fluctuations. In general, for each transmitting node $t \in \mathcal{R} \setminus \{D\}$ there are up to d subsequent nodes in the route that are overhearing. The cooperative set of nodes $\mathcal{T}_{k,d}$ that are transmitting towards the terminal k as part of the cooperative link $(\mathcal{T}_{k,d}, k)$ are thus defined as $\mathcal{T}_{k,d} = \{k - d, ..., k - 1\} \subset \mathcal{R}$. In [14] it is proposed the use of a switched combining (see also Fig. 2(d)) scheme for processing of multiple RSSs obtained from the cooperative link.

B. Global data sharing: decentralized consensus

Global data sharing allows distributed cloud devices to reach a consensus on an estimate of a physical quantity. Rather than letting the Host station to collect all the measurements (e.g., acting as fusion center) for centralized estimation, decentralized consensus methods allow cloud devices to obtain the estimation of the physical quantities of interest without the need to forward measurements to the Host station, nor exchange measurements among peer devices. The cloud devices can reach a consensus on any physical parameter of interest by iterative refinements and exchange of local noisy estimates [9]. In the example of Fig. 3, we assume that vector

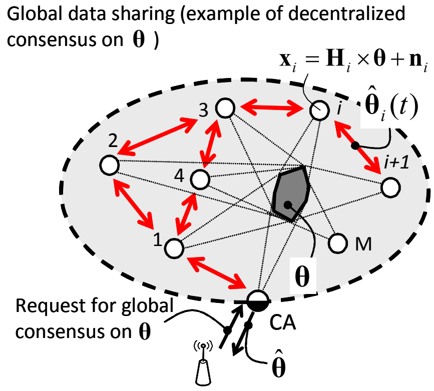


Fig. 3. Global data sharing for distributed estimation of parameter $\boldsymbol{\theta}$

$\boldsymbol{\theta} = [\theta_1, ..., \theta_p]^T$ contains p parameters representative of the physical quantity to be estimated (i.e., a list of parameters describing the network or plant state). Each k-th cloud device collects a set of T temporal samples $\mathbf{x}_k = [x_k(1), ..., x_k(T)]^T$ of noisy local observations

$$\mathbf{x}_k = \mathbf{H}_k \times \boldsymbol{\theta} + \mathbf{n}_k \tag{1}$$

where $\mathbf{n}_k$ accounts for random zero-mean instrumentation noise with term $\mathbb{E}[\,|\mathbf{n}_k|^2\,] = \mathbf{N}_i$ and $\mathbf{H}_k$ is the linear regressor matrix. Vector $\mathbf{x}_k$ provides a local (and noisy) measure of the network or plant state from the point of view of device k. Let $\hat{\boldsymbol{\theta}}_k(t)$ be the local estimate of $\boldsymbol{\theta}$ at time t available at node k, then the iterative exchange and update of $\hat{\boldsymbol{\theta}}_k(t)$ for all cloud devices $k = 1, ...M$ at any time $t \geq 0$ yields asymptotically a consensus on the same value,

$$\hat{\boldsymbol{\theta}} = \lim_{t\to\infty} \hat{\boldsymbol{\theta}}_k(t)\,\forall k \tag{2}$$

and such that $\forall(k,j)\ \lim_{t\to\infty}\left[\hat{\boldsymbol{\theta}}_j(t) - \hat{\boldsymbol{\theta}}_k(t)\right] = 0$. This is obtained by applying the consensus iteration update equation for $t \geq 0$ as [15]

$$\hat{\boldsymbol{\theta}}_k(t+1) = (1-\epsilon_k)\hat{\boldsymbol{\theta}}_k(t) + \epsilon_k \sum_{j\in\mathcal{T}_{k,d}} \mathbf{W}_j \times \left[\hat{\boldsymbol{\theta}}_j(t) - \hat{\boldsymbol{\theta}}_k(t)\right] \tag{3}$$

with $\mathbf{W}_j$ being the weighting term for device $j \in \mathcal{T}_{k,d}$ communicating with node k as part of the cooperative link $(\mathcal{T}_{k,d}, k)$. Initial local estimator $\hat{\boldsymbol{\theta}}_k(0)$ of $\boldsymbol{\theta}$ is carried out at time $t = 0$ using the local measurements $\mathbf{x}_k$. For $p \leq T$ (see [9] for undetermined $p > T$) the best linear unbiased estimator (BLUE) is given by

$$\hat{\boldsymbol{\theta}}_k(0) = \mathbf{C}_k \mathbf{H}_k^T \mathbf{N}_k^{-1} \mathbf{x}_k \tag{4}$$

with $\mathbf{C}_k = (\mathbf{H}_k^T \mathbf{N}_k^{-1} \mathbf{H}_k)^{-1}$ being the initial local estimator covariance. Weighting terms $\{\mathbf{W}_k\}_{k=1}^M$ are chosen during network set-up, in addition these can be reasonably assumed as fixed and available to the cloud devices involved in the consensus process. Optimum weighting can be chosen as $\mathbf{W}_j = \mathbf{C}_j^{-1}\left(\sum_{h\in\mathcal{T}_{k,d}} \mathbf{C}_h^{-1}\right)^{-1}$, so that the lower is the covariance of the local estimate from neighbor node j, compared to covariances $\mathbf{C}_{h\in\mathcal{T}_{k,d}}$ of other neighbors, the higher is the reliability given to the corresponding received estimate $\hat{\boldsymbol{\theta}}_j(t)$. Other solutions are analyzed in [15]. The comparative analysis of global consensus $\hat{\boldsymbol{\theta}}$ (2) and corresponding local estimation $\hat{\boldsymbol{\theta}}_k(0)$ (4) can be used by any node k to enable early detection and localization of critical conditions.

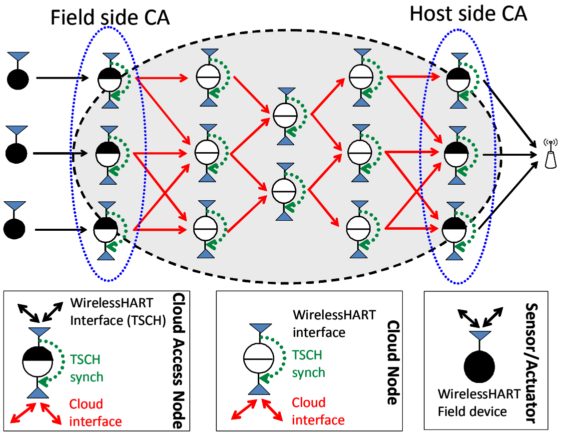


Fig. 4. Cloud-based WirelessHART: dual-layer implementation. Virtual communication between Host-side and Field-side CA pairs is implemented by the WCN cloud section.

IV. DUAL-LAYER ARCHITECTURE AND IMPLEMENTATION

The WirelessHART (IEC 62591) standard implementing the TSCH specifics will be more deeply considered in the following for the cloud-based network implementation. As depicted in Fig. 4 it is proposed a layered network architecture where the wireless cloud devices are equipped with a dual radio access technology (RAT) operating over the same 2.4GHz band. The first radio (WierelessHART layer) supports the WirelessHART standard and is adopted to synchronize the network to the reference clock provided by the network manager. The second radio (cloud interface layer) will support the WCN MAC section. The cloud network is activated by the CA nodes as these can off-load traffic through the cloud. The WCN MAC low-level software and the WirelessHART system run in parallel using different IEEE 802.15.4 logical channels. The 'interface' between the 'cloud' section and the 'WirelessHART' section is obtained via a (wired) serial-line connection transferring Highway Addressable Remote Transducer (HART) messages. This ensures full compatibility between the cloud and the WirelessHART air interfaces.

The CA nodes are logically separated into categories, namely 'host-side' cloud access nodes (CAH) providing to the Host station an access point to the cloud, and 'field-side' cloud access nodes (CAF) granting new requests for cloud resources coming from the HART end devices. The cloud should therefore make possible a 'virtual' connection between a CAF node (representative of a HART field device) and the selected CAH node. In order to evaluate the application cases of Sect. 2, the cloud section provides specific 'services' to the

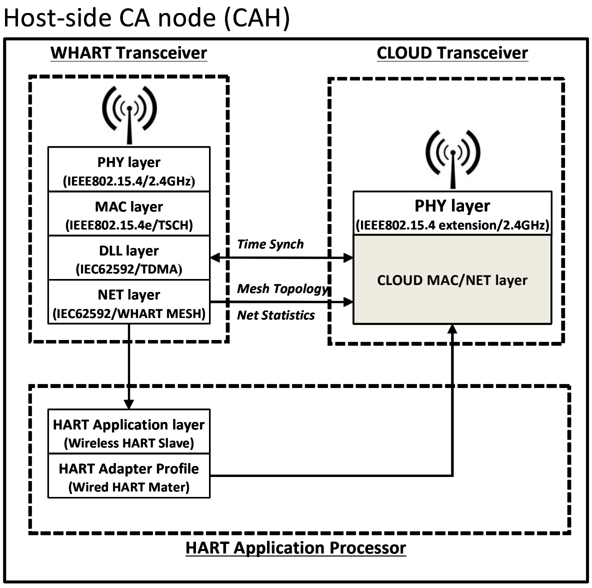


Fig. 5. Host-side CA node architecture.

HART section while a dedicated service will be associated to each application case.

A. Hardware and software architecture

The Radio Module for the cloud section supports the IEEE 802.15.4-2006 standard (PHY and MAC) including the 802.15.4e MAC amendments and the upper/application layer of the WirelessHART protocol. The high integration, System on Chip (SoC) solution from ATMEL (ATmega2564RFR2 device [11]) is found as reasonable choice after having analyzed the applicable requirements in the proposed context. The reference architecture for the identified dual-RAT solution would entail two independent 802.15.4 transceivers which are tightly coupled with a dedicated real-time processor such to support, in a coordinated way, the MAC layers associated with both the TSCH-based and the Cloud-based networks.

To ensure a better compatibility with the existing WirelessHART solution, it was decided to use independent processing modules (including both a CPU and an RF transceiver) each running the TSCH and the Cloud stack up to the Network Layer level. This independent modules will then communicate via a wired communication channel which – for uniformity reasons – will be based on the HART Transport layer format.

In what follows we detailed the different cloud network entities focusing on the specifics of the software architecture.

Cloud Access Nodes. The Cloud Access (CA) nodes provides to any standard – IEC62592 – WirelessHART device the access to the Wireless Cloud services. The *host-side* CA node (CAH) is a cloud access point that allows to any WirelessHART compliant Host station to access the cloud network services. The CAH HW/SW architecture is depicted in the Fig. 5. The WirelessHART transceiver acts as a wired master HART device application towards the Cloud network layer, with an approach similar to the what done within a WirelessHART Adapter device. The *field-side* CA node (CAF) is a cloud access point that allows to any standard WirelessHART field device to access the cloud network services. In contrast with the CAH node architecture, the Cloud transceiver acts as a master HART device towards the HART wired application layer, with an approach similar to the wired HART maintenance port available within each WirelessHART device.

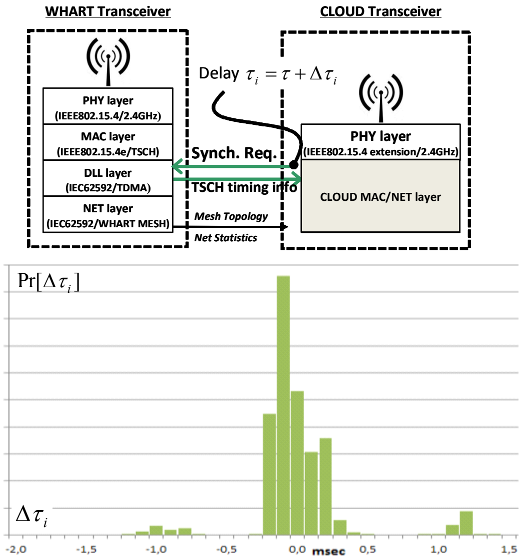


Fig. 6. Timing info acquisition from TSCH network.

Cloud Nodes (C) are the native building blocks for the wireless cloud itself. The WirelessHART transceiver and the Cloud transceiver protocol layers are interconnected so to exchange “mesh topology” and “synchronization” information. By collecting the neighbor information – as well as the associated statistics – the Cloud transceiver can get a local view of the network topology and also estimate the link quality level exploiting consensus based techniques. Within the WirelessHART network, a neighbor is defined as an “adjacent” node when its receive signal level suggests that communication in at least one direction is possible. In the following experiments, the signal quality is measured in terms of received signal strength indication RSSI - in dBm - and link quality indication LQI [16].

B. TSCH-assisted synchronization

The WirelessHART network time is defined by the “Absolute Slot Number” (ASN) value, that is kept synchronized within the HART-specified accuracy level (of $4ms$). A TSCH-assisted cloud synchronization scheme is thus implemented by cloud devices by periodically collecting ASN information from the WirelessHART network TDMA DLL layer through a request/response procedure. Considering that no standardized way is available to directly access the ASN information, the Cloud transceiver is designed to use the standard HART

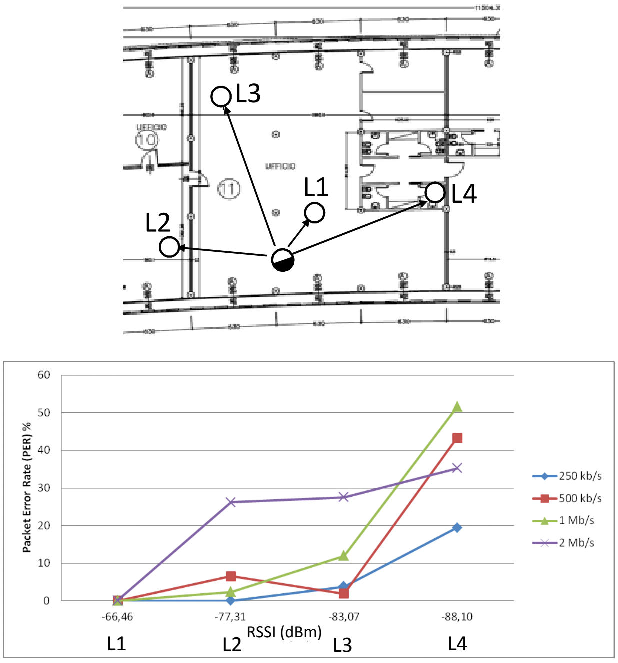


Fig. 7. Top. Locations L1-L4 of the cloud devices and CA node. Bottom. Packet Error Rate vs RSSI for different IEEE 802.15.4 data-rates (for positions L1-L4).

real-time clock command (CMD 90). For a WirelessHART device the real-time clock is internally derived from the ASN value, therefore it is possible – by suitable conversion and delay-compensation techniques – to reconstruct the original ASN information with an acceptable degree of accuracy. An experimental assessment of jitter distribution $\Delta\tau_i$ after compensation of predicted average request/response delay τ_i experienced by cloud device i is illustrated in Fig. 6. The maximum observed jitter is limited to $1ms$ so that the overall synchronization error inside the cloud network layer can be reasonably limited to $5ms$.

C. *IEEE 802.15.4 high data-rate mode*

This section describes two experiments aimed at analyzing the transceiver capabilities of the cloud node at the four different data rates available ($R_0 = 250kbps$, $R_1 = 500Kbps$, $R_2 = 1Mbps$, $R_3 = 2Mbps$). The use of the enhanced data-rate mode available in recent low-power IEEE 802.15.4 compliant transceiver devices [11] can be a promising option for fast servicing of high-priority cloud traffic. High-data rate transfer mode can be implemented by programming the devices to reduce the IEEE 802.15.4 direct sequence spread spectrum (DSSS) factor down to a value of $Q < 8$, corresponding to a PHY data rates of $500kbps$ (for $Q = 4$), $1Mbps$ (for $Q = 2$) and $2Mbps$ (for $Q = 1$).

In the first experiment, we measured the average Packet Error Rate (PER) between a transmitter and a receiver in different indoor locations (as depicted in Fig. 7 on top) at every data rate. Both the RSSI and the LQI are first measured when the cloud devices (acting here as receivers) are put in different indoor locations. In Fig. 7 at bottom we analyze the correlation between the signal strength and the average Packet Error Rate (PER) at the four data-rates. While the CA was fixed near the Host PC, the cloud devices were put in different indoor locations (L1 to L4) chosen in such a way to obtain a wide variety of RSSI levels. For each data rate and each receiver location, the CA node (acting here as transmitter) sends probe signals to cloud devices. When the cloud device receives a packet, it sends a response to the transmitter with the values of the receiver-side RSSI and LQI in the appropriate fields. The CA then notifies the Host (implemented in this tests on a PC) about the RSSI and LQI levels for that packet. The cloud devices can potentially change the data rate in real-time. Each transmitted packet contains a field that notifies the data-rate update for the next one: this is used by the CA node to modify its sensitivity threshold.

The plot in Fig 7 shows the PER as a function of the RSSI for each data rate. While the curves for $250kbps$ and $1Mbps$ are monotonically increasing as expected, the other two exhibit more unusual behaviors which are probably due to obstructions and multipath effects that cannot be captured by the RSSI alone. It can be also noted that for data-rates of $2Mbps$ (this case corresponds to disabling DSSS) the PER considerably increases for RSSI above -77 dBm, while for the remaining data-rate options performance degradation occurs between -83 dBm and -88 dBm.

In the second experiment, we measured the maximum throughput achievable at every data rate when sending a packet through a two-hop path (a routing path between the CAH and CAF node, see Fig. 8 on top). The goal of this experiment is to determine the maximum data throughput achievable in a two-hop round-trip route (with total hop count of 4) at different data rates and estimate the minimum amount of time that the software needs in order to process a packet. This analysis is thus crucial to assess the performance of hybrid multi-hop and cooperative incremental relaying schemes (see Sect. 3.1). The software on every node is based on a customized version of the light-weight mesh stack [11] and it is designed in such a way that it performs only a minimum number of operations between the reception of a packet and the transmission of the next one. Transmission sessions are also configured in such a way to disable acknowledgement option (implicit acknowledgement). In the proposed setting a CAH node sends a packet to a CAF device through an intermediate cloud device serving as store and forward node. When the CAF node receives the packet, it sends the response back to the CAH through the same cloud device. As soon as the CAH node receives the response, it start a new multi-hop session, sending the next packet. The CAH also notifies the host PC about the number of packets transmitted and received. The throughput computed by experiments is compared with the theoretical maximum throughput that could be achieved without packet drops and assuming that all the time is used

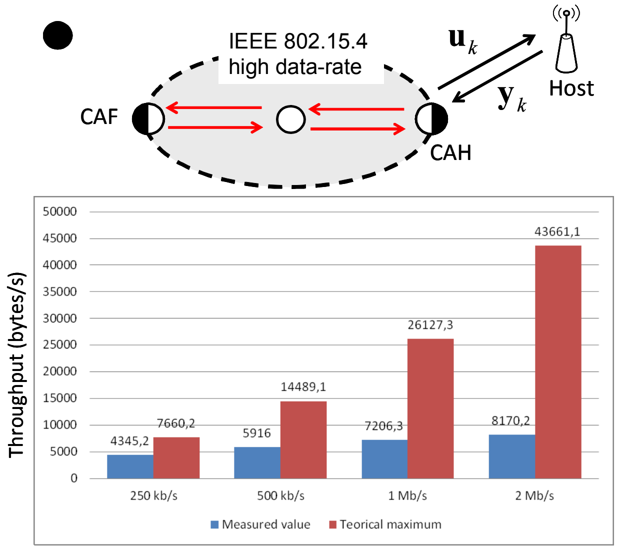


Fig. 8. Top. Multihop transmission testing with high-data rate IEEE 802.15.4 devices. Bottom. Theorical and observed throughput for two-hop cloud session.

to transmit and receive (no processing for decoding/storage of data frames). For IEEE 802.15.4 standard data-rate case with $R_0 = 250kbps$, the maximum throughput for a $N = 4$ multihop session can be computed as $T_0 = (1/N) \times PR_0/(H+P)$ where H is the size of the physical header (6 bytes) and P is the size of the frame payload (102 bytes in this case). For higher data rates, the physical header and preamble are always transmitted at data rate R_0 to allow for PHY layer synchronization: the throughput T_i for data rate $R_i > R_0$ with $i = 1, 2, 3$ can be thus written as

$$T_i = \frac{P}{N} \times \frac{R_i R_0}{HR_i + PR_0}. \tag{5}$$

The results of the experiments are shown in Fig. 8 at bottom. To allow a fair comparison with theoretical throughput the experimental setting is chosen so that the probability of packet drops is minimized. At the lowest data rate of R_0, the measured throughput is about 56,7% of the theoretical maximum, so the processing time is less than the transmission time. Compared to IEEE 802.15.4 standard data rate R_0, the observed throughput increase of 36% at data rate $R_1 = 500kbps$, 65% at rate $R_2 = 1Mbps$ and 88% at maximum rate of $R_3 = 2Mbps$. Therefore as the data rate increases, the throughput increases as well, but much slower than the theoretical value. At the maximum data rate of $R_3 = 2Mbps$, the actual measured throughput is 18,7% of the theoretical one (although it can be reasonably assumed that MAC software optimizations may provide improvements of efficiency at higher data rates). Given the reduced sensitivity of $2Mbps$ transmission mode due to the disabling of DSSS and the limited processing power of the integrated SoC, a reasonable option for development would be the use of intermediate data-rates (R_1,R_2).

V. CONCLUDING REMARKS

In this paper we propose an hardware and software architecture integrating a cloud network system with a wireless industrial sensor network compliant with the Time-Synchronized Mesh Protocol (TSCH) standard and WirelessHART (IEC 62591). Cloud devices implement enhanced IEEE 802.15.4 PHY data-rate transmission, and are configured to support cooperative-multihop transmissions and decentralized estimation by distributed consensus of critical plant parameters. Preliminary experimental measurements are carried out to evaluate the feasibility and the effectiveness of the proposed system, looking in particular at cloud synchronization and throughput in practical settings.

REFERENCES

[1] D. Zuehlke, “Smart factory - towards a factory-of-things,” Annual Reviews in Control, vol. 34, no. 1, pp. 129-138, April 2010.

[2] D. Miorandi, S. Sicari, F. De Pellegrini, I. Chlamtac, “Internet of things: vision, applications and research challenges,” Elsevier Ad Hoc Networks, vol. 10, Issue 7, pp. 1497-1516, September 2012.

[3] S. Petersen, S. Carlsen, “WirelessHART versus ISA100.11a: the format war hits the factory floor,” IEEE Ind. Electronics Mag., vol. 5, no.4, pp. 23,34, Dec. 2011.

[4] F. Boccardi, et al., “Five disruptive technology directions for 5G,” IEEE Comm. Magazine, Feb. 2014.

[5] A. Alamri, et al., “A survey on sensor-cloud: architecture, applications, and approaches,” International Journal of Distributed Sensor Networks, Hindawi Publ., vol. 2013, Article ID 917923, 18 pages, 2013.

[6] IEEE Standard for Local and metropolitan area networks - Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer, IEEE Std 802.15.4e$^{\mathrm{TM}}$-2012, April 2012.

[7] R. Piyare1 and S. R. Lee, “Towards Internet of Things (IoTs): integration of wireless sensor network to cloud services for data collection and sharing,” International Journal of Computer Networks & Communications (IJCNC) vol 5, no.5, September 2013.

[8] J. Sýkora and H. Mark, “Dense Cooperative Wireless Cloud Network (DIWINE)”, in Proceedings of the Future Network and Mobile Summit (FuNeMS), 3–5 July 2013.

[9] A. Bolognino, U. Spagnolini, “Consensus based distributed estimation with local-accuracy exchange in cloud-wireless systems,” Proc. of IEEE International Conf. on Comm. (ICC 2014), Australia, Sydney, June 2014.

[10] S. Savazzi, M. Nicoli, F. Carminati, M. Riva, “A Bayesian approach to device-free localization: modeling and experimental assessment,” IEEE Journal on Sel. Topics in Signal Proc., vol. 8, no. 1, pp.16-29, Feb. 2014.

[11] Atmel, ATmega256RFR2, 8-bit AVR microcontroller with low-power 2.4GHz transceiver for ZigBee and IEEE 802.15.4, Revision B, Feb. 2013.

[12] V. Nguyen, D. Brunelli, “Cooperative transmission range doubling with IEEE 802.15.4,” Proc. of IEEE International Conference on Communications, pp.126-130, June 2012.

[13] B. Zhao, M. C. Valenti, “Practical relay networks: a generalization of hybrid-ARQ,” IEEE Journal on Sel. Areas in Comm., vol 23, no. 1, pp.7-18, Jan. 2005.

[14] S. Savazzi, “Wireless virtual multiple antenna networks for critical process control: protocols and experiments” Hindawi, International Journal of Distributed Sensor Networks, vol. 2013, Article ID 973621, p.15, 2013.

[15] M. Nicoli, G. Soatti, S. Savazzi, “Distributed estimation of macroscopic channel parameters in dense cooperative wireless networks” Proc. of IEEE Wireless Communications and Networking Conference (WCNC 2014), Istanbul, Turkey, April 2014.

[16] N. Baccour et al. “Radio link quality estimation in wireless sensor networks: a survey,” Journal ACM Transactions on Sensor Networks (TOSN), vol. 8 Issue 4, no. 34, Sept. 2012.