Load Balancing for Reliable Self-Organizing Industrial IoT Networks

M.Carmen Lucas-Estañ, *Member, IEEE*, Javier Gozalvez, *Senior Member, IEEE*

***Abstract*—Industry 4.0 will interconnect and digitalize traditional industries to enable smart and adaptable factories that efficiently utilize resources and integrate systems. A key enabler of this paradigm is the communications infrastructure that will support the ubiquitous connectivity of Cyber-Physical Production Systems. The integration of wireless networks will facilitate the dynamic reconfiguration of the factories of the future, and the collection and management of large amounts of data. This vision requires reliable and low latency wireless links with the necessary bandwidth to support data intensive applications and spatio-temporal variations of data resulting from the reconfiguration of Industrial IoT (Internet of Things) systems. To this aim, this paper proposes a load balancing scheme that dynamically manages the wireless links based on their quality and the amount of data to be transmitted by each node. The proposed scheme avoids the saturation of channels, and significantly augments the reliability of industrial wireless networks in comparison with existing solutions.**

***Index Terms*—Industrial wireless networks, Industrial IoT, IIoT, industrial wireless sensor networks, Industry 4.0, Factories of the Future, self-organizing, load balancing, CPPS.**

I. Introduction

THE digitalization of the industry will introduce a significant number of changes to manufacturing processes, operation and systems. All these transformations are defined under the concept of Industry 4.0 [1]. Industry 4.0 is based on the interconnection and digitalization of traditional industries (such as manufacturing) to enable smart and adaptable factories that efficiently utilize resources, and integrate components and systems [2]. A key enabler of the Industry 4.0 paradigm is the communications infrastructure that will support the ubiquitous connectivity of Cyber-Physical Production Systems (CPPS) [3]. The adoption of wireless communications for such connectivity will enhance the flexibility and reconfiguration capability sought for Industrial IoT (IIoT) networks.

Industrial CPPS systems will be supported by different types of wireless sensors that can be fixed or mobile; mobile sensors can be associated to workers, mobile machinery or vehicles. These sensors will send data to control centers that will be in charge of controlling and supervising the industrial environment and manufacturing processes. The sensors can be of different nature, and have different communication requirements. For example, temperature sensors will transmit small amounts of data (usually periodically), while IP cameras or 3D scanners sporadically generate large amounts of data that require high bandwidth communication links. Traditional industrial wireless standards such as WirelessHART or ISA100.11a can only support low bandwidth data transmissions. Several studies have then proposed to support future industrial CPPS systems with hierarchical communication networks ([4]-[8]) that integrate and exploit various wireless technologies with different characteristics. In such hierarchical networks, sink nodes are deployed to collect data from different low-bandwidth sensors, and transmit it to gateway nodes using wireless technologies with higher bandwidth. The gateway nodes are usually deployed so that they can collect and transmit data from/to various sensors and/or sink nodes.

Hierarchical communication networks must be able to support dynamic Industrial IoT environments that result from the coexistence of different types of sensors, including mobile sensors in robots, machinery, vehicles, or even workers. These sensors can have varying data demands or generation rates that can result in spatio-temporal variations of the data demand and distribution within factories. Supporting these variations requires the capacity to dynamically manage the industrial wireless networks (IWNs). Such dynamic management is critical to ensure the reliability and self-organizing capability of the IWNs. In this context, this study proposes a novel load balancing scheme that is capable of dynamically reacting under changes in data demand and distribution in order to avoid the congestion of wireless links and the resulting loss of critical industrial data. In particular, the proposed scheme focuses on balancing the load of links between sink and gateway nodes since these links transmit large amounts of aggregated data. The load balancing decisions are based on the quality of the wireless links and the amount of data that each node must transmit. The conducted evaluation demonstrates that the proposed scheme reduces channel congestion and significantly improves the reliability of IWNs compared to existing solutions and static wireless deployments. The proposed scheme also reduces the number of reconfigurations of wireless links, and therefore the scalability and the

Manuscript received Sep. 3, 2018; revised Dec. 21, 2018; accepted Jan. 30, 2019. This work has been funded by the European Commission through the FoF-RIA Project AUTOWARE: Wireless Autonomous, Reliable and Resilient Production Operation Architecture for Cognitive Manufacturing (No. 723909). Paper no. TII-18-2305. (Corresponding author: M.C. Lucas-Estañ)

M.C. Lucas-Estañ and J. Gozalvez are with the UWICORE Laboratory, Universidad Miguel Hernandez de Elche (UMH), 03202 Elche, Spain (e-mail: m.lucas@umh.es, j.gozalvez@umh.es).

signaling overhead generated when deploying self-organizing IWNs. The main contributions of this paper are:

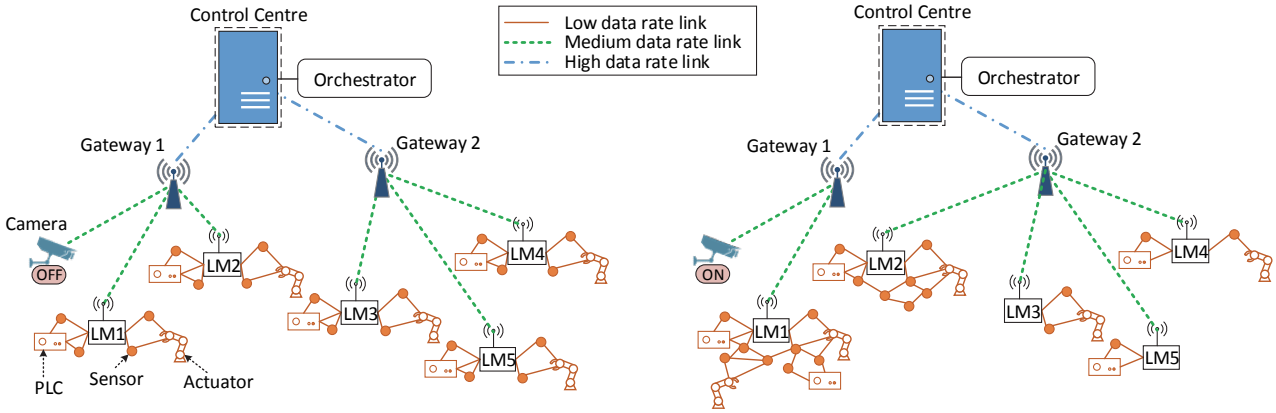
- The paper proposes a load-balancing scheme that improves the state of the art, and is capable to efficiently balance the load among gateway nodes serving multiple sink nodes in industrial wireless networks.
- The proposed scheme is based on a new metric that estimates the load that a channel experiences. The proposed metric uses existing information, in particular information about the quality of the wireless links and the amount of data that each sink node must transmit.
- The proposed scheme can operate under single and multi-channel scenarios.
- The paper conducts an exhaustive analysis that demonstrates that the proposed load balancing scheme outperforms existing solutions under multiple scenarios and operating conditions.
- The paper also demonstrates that the proposed load-balancing scheme guarantees a stable network operation with low overhead. In particular, the proposed scheme can better handle spatio-temporal variations of data in industrial IoT networks while limiting the number of reconfigurations of wireless links.

II. Industrial Wireless Networks

WirelessHART, ISA100.11a and IEEE802.15.4e are some of the existing standards for industrial wireless communications. These standards adopt the IEEE 802.15.4 physical layer and extend the capabilities of the IEEE 802.15.4 MAC (Medium Access Control Layer) layer to support a high number of field devices (sensor or actuators) that require low data rates and energy consumption. In general, these standards centrally manage the network to ensure reliable industrial wireless communications[1]. However, a centralized network management can result in excessive overhead, long reconfiguration times and scalability challenges ([4], [10]). To address these limitations, several studies (e.g. [4]-[8]) have proposed to deploy hierarchical IWNs capable of integrating multiple sub-networks supported by different wireless technologies that offer different connectivity capabilities. Each sub-network has its own manager and sink nodes. The manager manages the wireless connections of the sub-network, and the sink nodes collect/distribute the data in the sub-network. This paper considers that the manager and the sink nodes of a sub-network are implemented in the same physical node that is referred to as Local Manager (LM)[2]. LM nodes are connected to gateway nodes in the plant that aggregate data from different LM nodes, and transmit it to remote or on-site control centers/servers. Fig. 1.a represents an example of a hierarchical IWN following [7].

Several studies (e.g. [5]-[6]) have demonstrated that the reliability, delay and energy consumption of industrial networks can be improved when deploying heterogeneous wireless technologies capable of supporting different communication requirements (e.g. in terms of bandwidth, reliability or communication range). Such deployment is illustrated in Fig. 1.a. For example, WirelessHART, ISA100.11a and IEEE 802.15.4e can be utilized to support and manage sub-networks of sensors and actuators with low data rates. IEEE 802.11 (WiFi) or cellular technologies provide significantly higher bandwidth than existing industrial wireless standards, and their integration in industrial environments could be key to support the development of the Industry 4.0 paradigm. In fact, several studies have recently demonstrated the potential of IEEE 802.11 ([11]-[13]) and cellular technologies ([14]) to support industrial applications. The bandwidth of WiFi and cellular technologies make them suitable candidates to connect various LM nodes to Gateway nodes. These technologies could also be used to directly connect sensors that require high data rates (e.g. video cameras) to Gateway nodes as illustrated in Fig. 1.a. The Gateway nodes can be connected to remote or on-site control and data centers/servers using large-capacity (fixed or wireless) backhaul links.

Several studies have demonstrated the benefits provided by hierarchical industrial networks (e.g. [4]-[7]). These networks



a) Initial configuration.

b) Reconfiguration under changes in the spatial distribution of data.

Fig. 1. Hierarchical architecture in industrial wireless networks [7].

[1] Distributed management schemes are also considered in IETF 6TiSCH [9] in order to increase the flexibility and reduce the signaling overhead.

[2] A LM is equivalent to a Network Manager in WirelessHART or a System Manager in ISA100.11a.

can play a significant role in the development of the Industry 4.0 if they are able to support data-intensive applications and the foreseen spatio-temporal variations of data demand and distribution in factories. Such variations can challenge the reliable, timely and efficient transmission of data, and require flexible and agile networks capable of dynamically reconfiguring the wireless connections. An example of this challenge is illustrated in Fig. 1. The initial configuration of the network (Fig. 1.a) is initially capable of adequately collecting all data at the LMs and transmit it to the Control Center through the Gateway Nodes. However, in Fig. 1.b, a higher number of sensor nodes are connected to LM1 and LM2, and the camera has been activated. All these changes significantly increase the load at Gateway 1 with the subsequent risk of saturating its channels and lose critical industrial data. To avoid this scenario, it is necessary that the network detects the spatio-temporal variation of the data, and reconfigures the network connections to avoid any possible link saturation. This is done in Fig. 1.b by balancing the load of the wireless links, and connecting LM2 to Gateway 2. This example illustrates the need for IWNs to embed load balancing schemes capable of monitoring the status of wireless connections, detect possible risks of channel saturation, and be able to effectively distribute the data load among the available wireless nodes.

III. Related Work

Load balancing schemes have been proposed for conventional cellular and wireless networks with the objective to improve the network performance. For example, [15] proposes a scheme that balances the load among access points or base stations in order to avoid saturating backhauls links in a heterogeneous cloud radio access network. The scheme utilizes more efficiently the resources, and hence improves the network performance. In [16], the authors propose a user association scheme for a cellular network with several small cells and an overlapping macro cell. The proposed scheme decides which cell should serve each user by solving an optimization problem designed to maximize the throughput experienced by all users. The study shows that the maximum throughput is achieved when the scheme is capable of distributing the load among the different cells.

To the authors' knowledge, the only study that analyzes the application of load balancing in IWNs was presented in [17]. In [17], devices wirelessly communicate with Access Points (APs) that are connected to a global controller through a wired backbone. The scheme presented in [17] distributes devices between APs in order to maintain the load at each AP equal to the average network load (a maximum deviation per AP is allowed). The load of an AP is estimated in [17] as the total bandwidth required by all devices connected to the AP with respect to the total bandwidth available at the AP. The proposed solution is evaluated considering that all links of the same AP (and in some scenarios, even of all APs) experience the same Packet Error Rate (PER). In addition, [17] does not take into account the link quality experienced by the different devices connected to an AP in order to estimate the load of the AP. This can be highly relevant since a device with poor link quality requires much more bandwidth to transmit a given amount of data than another one with much better link quality. An alternative metric for load balancing is the data queue length of a node ([18]-[19]). This metric measures the total amount of data that the node has yet to transmit. In [18], the authors propose a load balancing scheme that distributes the load between the nodes that interconnect a mesh network with fixed IP networks. The load balancing decision is based on their level of congestion. The level of congestion is estimated as the average data queue length of the node. The study presented in [19] proposes a congestion control mechanism to balance the load in a wireless sensor network. To this aim, source nodes probabilistically decide which nodes should forward their messages to a sink node based on the estimated data queue length of potential forwarding nodes. The data queue length of a forwarding node is estimated considering its current queue length, the queue length of the source node, and the observed packet drops. The studies in [18] and [19] found that load balancing schemes that take into account the nodes' level of congestion significantly improve the throughput and reduce the delay.

The data queue length of a node can be a good indicator of its level of congestion. However, it does not provide sufficient information about the bandwidth required by the node to transmit its data to the destination. To estimate such bandwidth, it is also necessary to take into account the link quality (and its variations) of the wireless connection. In this context, this study proposes a novel load balancing scheme for IWNs. The scheme is designed with the objective to support the spatio-temporal variations of data demands and distribution in factories of the future. The proposed scheme bases its load balancing decisions on a new metric that estimates the time the channel is utilized; this metric can be easily estimated by the nodes. This study demonstrates that the proposed metric and load balancing scheme significantly improve the reliability of IWNs compared to static network deployments and alternative solutions. In particular, the proposed scheme improves the percentage of data packets successfully delivered to the destination node, and reduces the rate at which wireless links need to be reconfigured in order to support the spatio-temporal variations of data demands.

IV. Framework

We adopt the hierarchical IWN architecture proposed in [7] and illustrated in Fig. 1. The Control Center includes an Orchestrator that manages the complete IWN [8]. The LM and Gateway nodes continuously monitor the link quality (in particular, the Signal to Noise Ratio, SNR) of all their links, and periodically report it to the Orchestrator. The LM and Gateway nodes also include in the reports information about the amount of data (in bps) received from other nodes, and that has to be transmitted to the Control Center. The Orchestrator uses these reports to manage and reconfigure all the network connections in order to ensure the reliable, timely and efficient collection and distribution of data in the factory. This study focuses on balancing the load among Gateways by

dynamically managing the connections between LM nodes and Gateways[3]. The links between LM and Gateway nodes are critical since the LM nodes aggregate and transmit the data collected from various sensors. This study considers the use of IEEE 802.11 (or WiFi) to wirelessly connect LM and Gateway nodes. The Gateway nodes act as APs, and utilize IEEE 802.11a with Point Coordination Function (PCF)[4] in order to manage the access to the channel of the attached LM nodes and to prevent packet collisions [11]. As a result, a Gateway's channel can be used to serve several LMs. This study considers that several channels can be used by each Gateway node. In addition, this study assumes that each LM node is in the communication range of at least two Gateways. This is highly realistic since the reliability levels demanded by industrial applications generally results in the need for redundancy in network deployments [20].

V. Load Balancing Proposal

The proposed load balancing scheme estimates the load experienced by a channel as the percentage of time that the channel is utilized by all the LMs it serves. This metric is here referred to as *CU* or Channel Utilization percentage, and the proposed load balancing scheme is referred to as CUBE (Channel Utilization Balancing schemE). CUBE is executed at the Orchestrator that uses information monitored and periodically reported by the LM and Gateway nodes. In particular, the LM nodes inform the Orchestrator of the amount of data (in bps) that they have to transmit to the Control Center, and of their link quality (in particular, the Signal to Noise Ratio or SNR) with the Gateway nodes under range. The Gateway nodes continuously measure the load of their channels, and periodically report them to the Orchestrator. The Orchestrator uses the information received from the LM and Gateway nodes to decide when CUBE has to be executed to balance the load between the channels of the Gateways. When CUBE is executed, the Orchestrator sends to the LM and Gateways nodes the instructions to reconfigure their links if LM nodes have to change their channel with their serving Gateway or even change their serving Gateway. Fig. 2 shows the interaction between the nodes that participate in the management of the LM-Gateways links. More detailed information about such interaction and the operation of CUBE is provided in the following sections.

A. Load Balancing

CUBE decides to which Gateway j and channel c should each LM i be attached; $i \in [1, L]$, $j \in [1, G]$ and $c \in [1, C_j]$, with L, G and C_j representing, respectively, the number of LMs and Gateways in the IWN, and the number of channels available in Gateway j. To this aim, CUBE seeks minimizing the maximum load of any channel, which is expressed as:

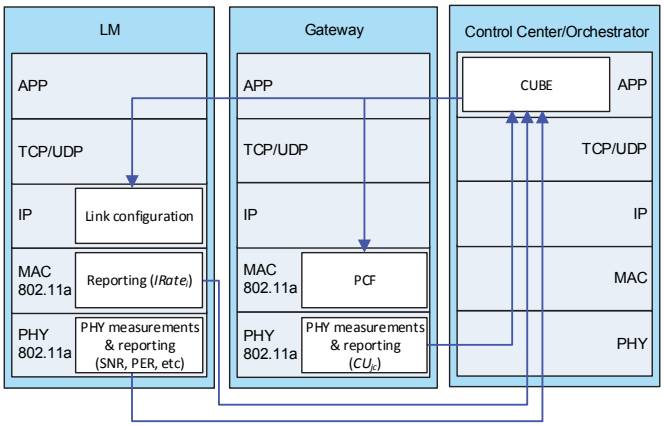


Fig. 2. Interaction between the nodes participating in the execution of CUBE.

$$\min \max_{j,c} \widehat{CU}_{jc}, \text{ where } \widehat{CU}_{jc} = \sum_{i=1}^{L} \widehat{CU}_{ijc} \cdot y_{ijc} \quad (1)$$

$\widehat{CU}_{jc}$ is the estimated load of channel c at Gateway j. $\widehat{CU}_{jc}$ can be expressed as the sum of the estimated load generated by each LM i served by the Gateway j using channel c ($\widehat{CU}_{ijc}$). y_{ijc} is a binary variable equal to 1 if LM i communicates with Gateway j using channel c, and equal to 0 otherwise. CUBE balances the load between channels in order to minimize the maximum load of any channel. The function in (1) can be expressed linearly as defined in (2), and considering the restriction expressed in (2.1). K is defined in (2.2).

$$\min K \quad (2)$$

$$s.t.: \quad \sum_{i=1}^{L} \widehat{CU}_{ijc} \cdot y_{ijc} \leq K, \ \forall j \in \{1, \ldots, G\}, \forall c \in \{1, \ldots, C_j\} \quad (2.1)$$

$$K \in \Re, K<1 \quad (2.2)$$

CUBE also prioritizes LMs changing the channel within the serving Gateway before changing to a different Gateway. This approach reduces the signaling and network overhead. To this end, CUBE seeks minimizing the following expression:

$$\min \sum_{j=1}^{G} \sum_{c=1}^{C_j} \sum_{i \in \Lambda_{j,c}} \sum_{\substack{m \in [1,G] \\ m \neq j}} \sum_{n=1}^{C_m} y_{imn} \quad (3)$$

In (3), $\Lambda_{j,c}$ represents the set of LMs attached to Gateway j using channel c. Let's consider that LM i is served by Gateway j using channel c, i.e. LM $i \in \Lambda_{j,c}$. Let's suppose that after executing CUBE, LM i is assigned to a different serving Gateway m with $m \neq j$ using channel n, $n \in [1, C_m]$; variable y_{imn} (with $m \neq j$ and $n \in [1, C_m]$) is equal to 1. Then, if LM $i \in \Lambda_{j,c}$ is assigned to a different Gateway node, the sum of all y_{imn} (with $m \in [1, G]$ and $m \neq j$, and $n \in [1, C_m]$) is equal to 1, as expressed in (4). Otherwise, the left-side expression in (4) is equal to 0. By solving (3), CUBE reduces the number of LMs that change their serving Gateway.

[3] The scheme could also be applied to manage the backhaul connections between the Gateways and the Controller, although the high bandwidth of these connections reduces the risk of channel saturation. The scheme can also be applied to manage the links between sensor and LM nodes. In this case, advertise messages or beacons sent by the sensor nodes could be used to estimate the link quality. Integration with routing protocols should be considered in the case of mesh network topologies.

[4] In PCF, an AP manages the access to the channel by sending polling messages to the attached nodes. Only the node that is addressed in a polling message can transmit at that time.

$$\sum_{\substack{m \in [1,G] \\ m \neq j}} \sum_{n=1}^{C_m} y_{imn} = 1 \quad (4)$$

CUBE solves then the following optimization problem:

$$o.f.: \quad \min K + \frac{1}{W} \sum_{j=1}^{G} \sum_{c=1}^{C_j} \sum_{i \in \Lambda_{j,c}} \sum_{\substack{m \in [1,G] \\ m \neq j}} \sum_{n=1}^{C_m} y_{imn} \quad (5)$$

$$s.t.: \quad \sum_{i=1}^{L} \widehat{CU}_{ijc} \cdot y_{ijc} \leq K, \ \forall j \in \{1, \ldots, G\}, \forall c \in \{1, \ldots, C_j\} \quad (5.1)$$

$$\sum_{j=1}^{G} \sum_{c=1}^{C_j} y_{ijc} = 1, \ \forall i \in \{1, \ldots, L\} \quad (5.2)$$

$$K \in \Re, K < 1 \quad (5.3)$$

$$W \in \Re, 0 < 1/W << 1 \quad (5.4)$$

$$y_{ijc} \in \{0, 1\} \quad (5.5)$$

The objective function is now defined in (5). The second term in (5) is multiplied by the factor $1/W$, where W is a large number (see restriction (5.4)). As a result, CUBE seeks minimizing K with the incentive to prioritize solutions that reduce the number of times that LMs change their serving Gateway. CUBE also guarantees that all LMs are connected to a Gateway following the restriction expressed in (5.2). The optimization problem is a mixed integer programming (MIP) problem with binary variables y_{ijc} and the real variable K.

CUBE is executed at the Orchestrator. Each Gateway continuously measures the load of its channels (i.e. CU_{jc}), and sends this information to the Orchestrator every t_{CUBE}. The Orchestrator periodically checks (every t_{CUBE}) for every Gateway j and channel c if CU_{jc} is higher than a predefined threshold CU_{th}. If it is the case, the Orchestrator executes CUBE to balance the load between the channels[5]. An adequate selection of CU_{th} is important to ensure that the load is balanced between channels while avoiding unnecessary signaling overhead due to frequent changes of serving Gateways. Small CU_{th} values can result in frequent (and possibly unnecessary) executions of CUBE, while large values may result in unbalanced load levels between the channels. The value of CU_{th} is then updated as a function of the optimum value of K (represented by K^*) after the last execution of CUBE. CU_{th} is updated so that CUBE can quickly react when the load is unbalanced between the channels of the different Gateways. Algorithm I shows how CU_{th} is updated. If CU_{th} is smaller than K^*, CU_{th} is updated to $K^*+\beta_1$; this ensures that CU_{th} is slightly higher than K^*. If CU_{th} is higher than or equal to K^*, CU_{th} is reduced by the factor β_2. It is not a problem if CU_{th} is temporarily higher than K^* since CUBE had previously assigned LMs to channels so that the Gateways can support higher channel load levels. However, CUBE is also executed if the time t_{ela} elapsed since its last execution is higher than T_{exe}[6] so that the channel load levels do not remain unbalanced for a significant amount of time. Fig. 3 summarizes the operation of CUBE.

ALGORITHM I: CU_{TH} UPDATE ($\{\beta_1, \beta_2\} \in \mathbb{R}$, $0 < \{\beta_1, \beta_2\} < 1$)

1. **If** $CU_{th} < K^*$
2. $CU_{th} = K^* + \beta_1$
3. **Else**
4. $CU_{th} = CU_{th} \cdot \beta_2$
5. **If** $CU_{th} < K^*$
6. $CU_{th} = K^* + \beta_1$
7. **End If**
8. **End If**

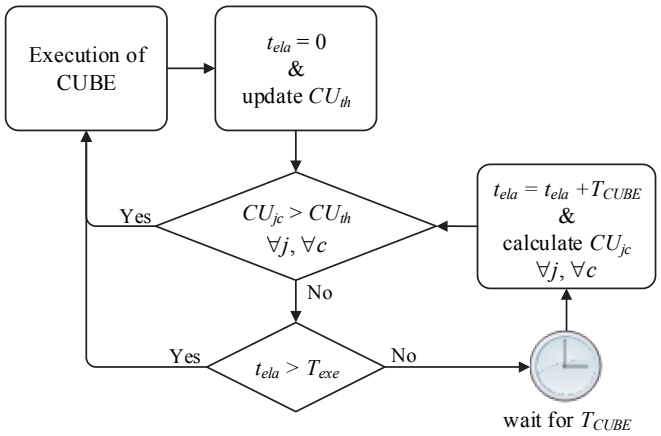


Fig. 3. Operation of CUBE.

B. Load Estimation

To estimate $\widehat{CU}_{ijc}$, each LM i measures the value of the SNR with each Gateway within its communication range. This is done using the Collision Free (CF)-Poll frames that are periodically transmitted by each Gateway following the IEEE 802.11a standard configured with PCF. The LMs report to the Orchestrator the SNR values every t_{CUBE}. The SNR is reported together with the rate $IRate_i$ at which data arrives at the MAC sublayer (from the upper layers) and enters the queue of the LM. We refer to this data as the data packet since this data forms the payload or MSDU (MAC Service Data Unit) of the packet that is finally sent through the radio link. With this information, the Orchestrator estimates the transmission rate ($ORate_{ijc}$) at which this data should be sent by LM i to each Gateway j under range when using channel c so that its data queue length (at the MAC level) does not augment. To this aim, LM i requires a transmission rate $ORate_{ijc}$ with Gateway j equal or higher than[7]:

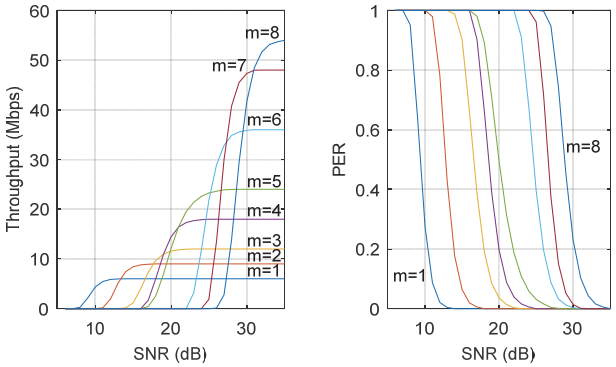
$$ORate_{ijc} \geq \frac{IRate_i}{1\text{-}PER_{ijc}} \quad (6)$$

[5] We also evaluated the scenario in which CUBE is executed periodically without observing significant performance benefits. On the other hand, periodically executing CUBE augments by a factor of 4 the number of times LM nodes change their serving Gateway.

[6] The values for β_1, β_2 and T_{exe} have been established experimentally and are shown in Table I. In particular, values have been selected to ensure an adequate tradeoff between minimizing data loss and reducing the number of times that LMs change their serving Gateways.

[7] $IRate_i$ and $ORate_{ijc}$ do not consider any headers or overhead bits added at the MAC and PHY layers.

where PER_{ijc} is the Packet Error Rate between LM i and Gateway j when using channel c. Satisfying (6) is important to prevent the loss of data as a result of the overflow of the data queue at the LMs. The Orchestrator estimates PER_{ijc} using the received SNR_{ijc} estimates (averaged over t_w) and the LUTs (Look Up Tables) illustrated in Fig. 4 (and derived from [21]). These LUTs relate the throughput and PER with the SNR for all possible transmission modes m included in IEEE 802.11a. A transmission mode is a combination of modulation and coding scheme. IEEE 802.11a defines 8 transmission modes, and each transmission mode has a different data rate R. In IEEE 802.11a, the transmitter dynamically selects the transmission mode m that maximizes the throughput for the experienced SNR. Using the average SNR_{ijc} estimate and Fig. 4.a, the Orchestrator identifies the transmission mode m_{ijc} that would maximize the throughput between LM i and Gateway j when using channel c. Once m_{ijc} has been identified, the Orchestrator can estimate PER_{ijc} using the average SNR_{ijc} and the LUT in Fig. 4.b.



a) Throughput as a function of SNR b) PER as a function of SNR

Fig. 4. LUTs for the different transmission modes in IEEE 802.11a [21].

We define L_{max} (in bits) as the maximum length of the payload or MSDU of a data packet transmitted through an IEEE 802.11a interface. To satisfy (6), LM i should transmit P_{ijc} data packets with a payload of L_{max} bits to Gateway j using channel c, plus one additional data packet with a payload of L bits per second. P_{ijc} and L are expressed as:

$$P_{ijc} = \left\lfloor \frac{ORate_{ijc}}{L_{max}} \right\rfloor = \left\lfloor \frac{IRate_i / (1 - PER_{ijc})}{L_{max}} \right\rfloor \quad (7)$$

$$L = \frac{IRate_i}{(1 - PER_{ijc})} - P_{ijc} \cdot L_{\max} \quad (8)$$

The Orchestrator can then estimate the value of $\widehat{CU}_{ijc}$ as:

$$\widehat{CU}_{ijc} = (P_{ijc}\text{-}1) \cdot \hat{T}_{ijc}(L_{max}) + \hat{T}_{ijc}(L) \quad (9)$$

$\hat{T}_{ijc}(L_{max})$ and $\hat{T}_{ijc}(L)$ represent the estimation of the time that LM i occupies channel c when it transmits to Gateway j a data packet with a payload of L_{max} and L data bits respectively. The PCF mode of IEEE 802.11a requires the transmission of a CF-Poll Frame from the Gateway to the LM before the LM can transmit a data packet to the Gateway. In addition, the LM must wait for t_{SIFS} (equal to 16 μs) after the reception of the CF-Poll Frame before it can start transmitting its data packet. The Gateway must also wait t_{SIFS} after it received the last data packet before transmitting another CF-Poll Frame. The time $\hat{T}_{ijc}(d)$ that LM i occupies channel c when transmitting a data packet with a payload of d data bits to Gateway j is then equal to:

$$\hat{T}_{ijc}(d) = t_{Poll} + t_{SIFS} + t_{PKT}(d) + t_{SIFS} \quad (10)$$

where t_{Poll} represents the time necessary to transmit a CF-Poll Frame, and $t_{PKT}(d)$ represents the time necessary to transmit a data packet with a payload of d bits of data. The time necessary in IEEE 802.11a to transmit a data packet with a payload of d bits of data is equal to:

$$t_{PKT}(d) = t_{PLCP-P} + t_{PLCP-H} + t_{MAC-H} + t_{payload}(d) + t_{FCS} + t_{tail} + t_{pad}. \quad (11)$$

where t_{PLCP-P} and t_{PLCP-H} represent the time necessary to transmit the preamble and PLCP (Physical Layer Convergence Procedure) header added in the IEEE 802.11a physical layer. t_{PLCP-P} and t_{PLCP-H} are equal to 16 μs and 4μs respectively. t_{MAC-H} and t_{FCS} represent the time necessary to transmit the 34 bytes added at the MAC layer, and that correspond to the MAC header and the Frame Check Sequence (FCS). $t_{payload}(d)$ represents the time necessary to transmit d data bits. Finally, t_{tail} and t_{pad} represent the time needed to transmit the tail bits and pad bits (16 and 6 bits respectively) that IEEE 802.11a adds to each packet prior to its radio transmission. If LM i uses the transmission mode m_{ijc} (with data rate $R(m_{ijc})$) to communicate with Gateway j using channel c, $t_{PKT}(d)$ is equal to:

$$t_{PKT}(d) = 20\,\mu s + \frac{34 \cdot 8 + 16 + 6 + d}{R(m_{ijc})} \quad (12)$$

The CF-Poll Frame contains a physical layer preamble and PLCP header, a data field of 20 bytes, and tail and pad bits (16 and 6 bits). The CF-Poll Frame packet is transmitted with the more robust transmission mode (corresponds to a data rate of 6 Mbps). In this context, t_{Poll} is equal to:

$$t_{Poll} = t_{PLCP-P} + t_{PLCP-H} + t_{payload}(20 \cdot 8) + t_{tail} + t_{pad} = 20\,\mu s + \frac{16 + 6 + 20 \cdot 8}{6 \cdot 10^6}. \quad (13)$$

Using (6)-(13), the Orchestrator can estimate the value of $\widehat{CU}_{ijc}$ that results from the transmission of LM i to Gateway j using channel c.

VI. Reference Schemes

The performance obtained with CUBE is compared in this study against that achieved with a static network deployment where each LM is permanently connected to the gateway with which it experiences the highest average SNR. This configuration is the most common in existing deployments, and is referred to in the rest of the paper as fixedGW.

CUBE is also compared to a load balancing scheme that bases its decisions on the data queue length of the LMs following the review of the state of the art presented in Section III. As discussed in Section III, several contributions (e.g. [18]

and [19]) utilize this metric for their load balancing proposals. This second reference scheme is referred to as QUEUE in the rest of the paper. For a fair comparison, QUEUE is implemented in this study following a similar approach to that considered for CUBE, but basing all decisions on the data queue lengths rather than on the $\widehat{CU}_{ijc}$ metric. In QUEUE, LMs also periodically send to the Orchestrator (every t_q) information about their maximum data queue lengths during the last t_q period. QUEUE calculates for each LM i the ratio QR_i between the maximum data queue length $QL_{max,i}$ experienced in the last t_q period and the capacity QC_i of its queue defined as the maximum amount of data that the queue can store:

$$QR_i = \frac{QL_{max,i}}{QC_i} \tag{14}$$

If QR_i is higher than a predefined threshold[8] QR_{th}, QUEUE assigns LM i a different Gateway or a different channel within the same serving Gateway if the following conditions are met: 1) all the LMs served by the new Gateway or the new channel must experience a value of QR below QR_{th}, and 2) LM i must have been served by the current Gateway for longer than t_{min}. This last condition is defined to avoid continuous changes of the channel or the serving Gateway. In fact, an LM that has recently changed its channel or its serving Gateway needs some time to reduce its QR below QR_{th}. If the two conditions are not satisfied for LM i, LM i maintains its current channel, and QUEUE tries instead changing the channel or the serving Gateway to the LM (different from LM i) that experiences the highest value of QR (even if it is lower than QR_{th}). Changing the channel or serving Gateway for this other LM can again only be executed if the two previous conditions are satisfied.

VII. Evaluation Platform and Scenarios

The schemes are evaluated using a custom discrete-event simulator developed by the authors in C++. The simulator implements all the relevant aspects necessary to accurately evaluate the performance of load balancing schemes in industrial wireless networks. In particular, the platform accurately models and simulates the LMs to Gateways connections[9] that implement the load balancing schemes under evaluation. This includes the MAC and PHY layers of IEEE 802.11a with its PCF function [22] used by the Gateway nodes to coordinate the access to the channel of different LMs. The LMs can be simultaneously connected with two Gateways to ensure the reliability of wireless connections. The simulator includes SNR maps (Fig. 5) to model radio propagation effects. These SNR maps have been obtained from real measurements (presented in [13]) in an industrial plant with wide corridors and large working areas typically separated by concrete walls. This plant is similar to the scenario simulated in this study and represented in Fig. 6. The corridors are machinery assembly areas, and typically present large metal pieces. The SNR map has been obtained from the measurements carried out at the 5.4GHz frequency band. These measurements accounted for varying operating and propagation conditions, including: Line Of Sight (LOS) with reduced obstructions; partial Non Line Of Sight (NLOS) due to cranes, pillars and machinery; and NLOS due to multiple obstructing elements or heavy obstructions. The SNR map represents the average SNR experienced by a node at distance (x, y) from an IEEE 802.11a transmitter located at the coordinates (0, 0). Fast-fading effects are included in the simulator through the use of LUTs that represent the physical layer performance (generally represented in terms of Packet Error Rates) as a function of the SNR. This simulator implements the LUTs presented in [21] for all the eight MCSs of IEEE 802.11a and considering a packet length of 1500 bytes. The simulator also implements a rate adaptation algorithm that dynamically selects the IEEE 802.11a transmission mode m that maximizes the throughput as a function of the average SNR. The simulator also includes the libraries and functions necessary to interact with IBM ILOG CPLEX [23] that has been used to solve the MIP problems defined in CUBE.

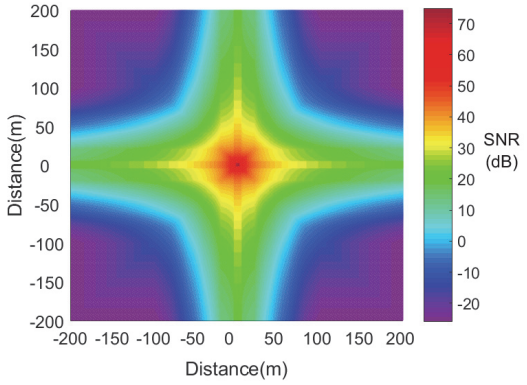


Fig. 5. SNR maps for 802.11a transmissions in industrial environments [13].

The schemes are evaluated in a scenario emulating an industrial plant of 300m x 200m with hallways that are 20m wide and that are distributed as illustrated in Fig. 6. This scenario is based on a real industrial plant consisting of wide corridors and large rooms separated by concrete walls or storage racks [13]. The scenario includes 3 Gateway and 9 LM nodes. This deployment guarantees wireless coverage in all the plant. In this scenario, each Gateway is assigned a single channel[10]. The scenario includes F fixed sensor nodes homogenously distributed in the plant. There are also M mobile sensor nodes (attached to mobile machinery, vehicles or workers, for example). Mobile nodes move following the Manhattan mobility model. When a node reaches an intersection, the probability of going straight, turning right and turning left is 0.5, 0.25 and 0.25 respectively. Nodes move

[8] The value of QR_{th} has been selected experimentally so that packet losses are reduced while controlling and diminishing the number of times that each LM changes its serving Gateway.

[9] How data is routed from a sensor node to the LM does not influence the operation of the load balancing schemes implemented at the LM-Gateway connections. We hence assume that each sensor (fixed and mobile) sends their data to the closest LM.

[10] The same performance trends (and gains from CUBE) have been observed when each Gateway is assigned one or multiple channels.

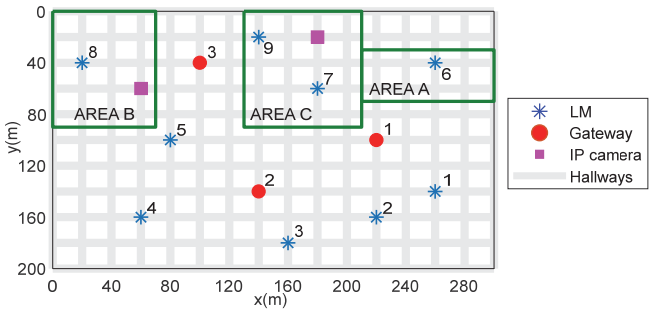


Fig. 6. Evaluation environment.

around the plant at a constant speed that is randomly chosen between 0.1 and 3m/s. All (fixed and mobile) sensor nodes transmit 10 packets (of 40 bytes each) per second[11]. Raw sensor data received at the LMs is converted to SensorML format before being forwarded to the Control Centre. This conversion increases the amount of data to be sent by a factor f equal to 10 as discussed in [7]. Table I summarizes the main simulation parameters.

Different scenarios have been simulated. In all scenarios, mobile sensor nodes can in principle move across the complete plant. However, these nodes tend to concentrate in certain areas of the plant when specific tasks or activities are executed in these areas. When these tasks are completed, mobile sensor nodes can again move freely across the plant. The three schemes have been thoroughly evaluated in scenarios S1 and S2 that differ on the duration and location of the tasks, and on the spatial distribution of the sensed data:

- Scenario S1. Tasks are concentrated in the areas A and B (Fig. 6). The tasks in A last from $T_{A,s}$ to $T_{A,e}$, and in B from $T_{B,s}$ to $T_{B,e}$. N_A and N_B mobile nodes move to areas A and B respectively during the execution of the tasks (Table II).
- Scenario S2: Tasks are concentrated in areas B and C (Fig. 6). The tasks in B last from $T_{B,s}$ to $T_{B,e}$, and those in C from $T_{C,s}$ to $T_{C,e}$. N_B and N_C mobile nodes move to areas B and C respectively during the execution of the tasks (Table II). In S2, IP cameras are switched during the execution of the tasks. The cameras produce video at a rate of 10 frames per second (see Table I). The presence of these cameras significantly increases the data load in the working areas compared to S1.

Additional simulations have been conducted in scenarios that modify some of the conditions or parameters of S1. The objective is to demonstrate that the benefits obtained with CUBE are not dependent on the configuration of the scenario or the simulation platform. The main characteristics of these additional scenarios are:

- Scenario S3: This scenario models the radio propagation effects using the model presented in [24] for industrial environments instead of the SNR maps. This model considers a one-slope log-distance path-loss, and takes into account the shadow fading effects through a log-normal random distribution with median equal to 0 dB. This model considers LOS and NLOS conditions (large-scale fading topographies 1 and 3 in [24]), and the model parameters configured in the simulation platform correspond to the 5.2 GHz model for non-fixed intercept.
- Scenario S4: This scenario considers a diverse industrial data traffic scenario thanks to the deployment of temperature sensors (TS), acceleration sensors (AS) and mobile robots (MR) that transmit 100 packets per second. The size of the packets is equal to 32 bits, 100 bits and 40 bytes for the TS, AS and MR respectively. These traffic patterns are representative of typical industrial applications for the Factory of the Future following [25].
- Scenario S5: This scenario introduces a non-homogenous distribution of the sensor nodes. In particular, 30% of the fixed and mobile nodes are located in Area D at the start of the simulation, 20% of the fixed and mobile sensor nodes are located in Area E, and 50% of the fixed and mobile sensor nodes are distributed homogeneously outside these two areas. Areas D and E are shown in Fig. 7.

[11] This traffic pattern is representative, for example, of data generated by sensors of a packaging machine, messages exchanged by a mobile control panel and a PLC, or control messages exchanged between a mobile robot and a remote guidance control system [25].

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Size of packets transmitted by LM nodes	1500 bytes
Queue capacity (QC) of LM nodes	32 kbytes
Camera frame size	50 kbytes
Frames per second sent by cameras	10
Queue capacity (QC) of IP cameras	500 kbytes
t_w	1s
t_{CUBE}	0.2s
T_{exe}	30s
t_q	0.2s
SNR_{th}	15dB
QR_{th}	0.98
t_{min}	5s
Raw Sensor data to SensorML format conversion factor, f	10
β_1, β_2	0.05, 0.95
W	0.001

TABLE II
CONFIGURATION OF SCENARIOS

Scenario	Parameter	Value
S1, S5, S6, S7	F, M	400, 300
	$T_{A,s}$, $T_{A,e}$, N_A	110s, 500s,100
	$T_{B,s}$, $T_{B,e}$, N_B	100s, 500s, 200
S2	F, M	400, 300
	$T_{B,s}$, $T_{B,e}$, N_B	150s, 700s, 100
	$T_{C,s}$, $T_{C,e}$, N_C	100s, 500s, 100
S3	F, M	450, 450
	$T_{A,s}$, $T_{A,e}$, N_A	110s, 500s,150
	$T_{B,s}$, $T_{B,e}$, N_B	100s, 500s, 300
S4	F (MR, TS, AS), M (MR, TS, AS)	200 (0, 100, 100), 180 (90, 45, 45)
	$T_{A,s}$, $T_{A,e}$, N_A	110s, 500s, 60 (30, 10, 20)
	$T_{B,s}$, $T_{B,e}$, N_B	100s, 500s, 80 (40, 20, 20)

- Scenario S6: This scenario introduces different working areas to those defined in S1: areas A and B in Fig. 7 (area B in S6 is the same as area D in S5). The scenario maintains the same duration of the tasks in these areas as S1, as well as the number of nodes that move towards these areas during the duration of the tasks.
- Scenario S7: S7 changes the deployment of the Gateway nodes as illustrated in Fig. 7.

Table II lists the main scenario parameters.

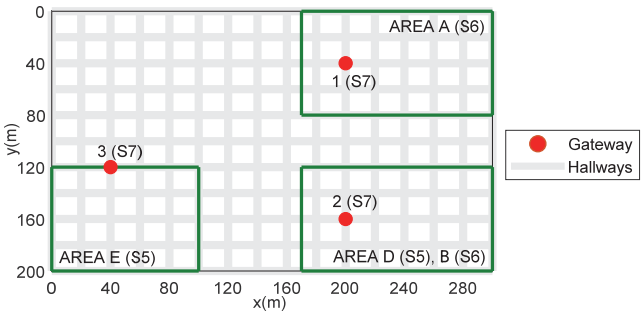


Fig. 7. Modified evaluation environment for S5, S6 and S7.

VIII. Performance Analysis

Fig. 8 depicts the percentage of lost packets for the baseline scenarios S1 and S2, and the three schemes under evaluation: a fixed assignment of LM to Gateway ('fixedGW'), and the QUEUE and CUBE load balancing schemes. As previously indicated, IEEE 802.11 is used in this study to wirelessly connect LM and Gateway nodes. The Gateway nodes act as APs, and utilize IEEE 802.11a with PCF in order to manage the access to the channel of the attached LM nodes. To this end, a Gateway sends polling messages to the attached LMs. Only the LM that is addressed in a polling message can transmit at that time. By using PCF, packet collisions can then be prevented. As a result, packet losses mainly result from the overflow of the data queues of the LMs. An overflow can occur if the channel serving the LM is overloaded, and hence the LM node cannot access the channel the time needed to transmit all the buffered data. Fig. 8 presents results only for those LMs that experienced a non-negligible number of errors[12]. In particular, the results are depicted for LMs number 5, 6 and 8 in S1, and number 5, 7 and 8 in S2. Their location is depicted in Fig. 6. These LMs correspond to those deployed inside or close to the working areas specified in Fig. 6 for scenarios S1 and S2. These areas can concentrate a higher number of nodes during the execution of the tasks, and hence the network load increases. Fig. 9 illustrates how all the data transmitted by the sensor nodes in the plant is distributed among the LMs. In particular, the figure represents the percentage of the total data generated by the sensor nodes that is managed by each LM. The colors in Fig. 9 are used to indicate the Gateway to which each LM is attached in the case of fixedGW. Fig. 8 and Fig. 9 show that the LM nodes that experience the higher packet losses are those that receive the

[12] Packet losses were almost equal to zero for LMs not represented in Fig. 8.

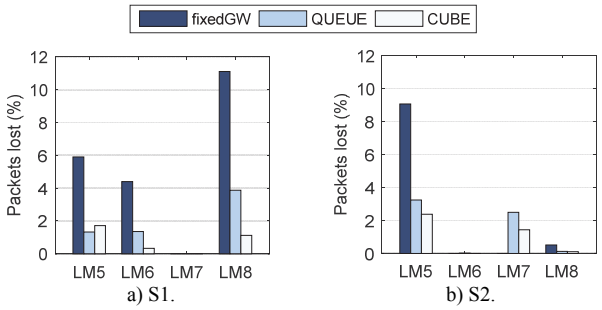


Fig. 8. Percentage of lost packets at different LMs.

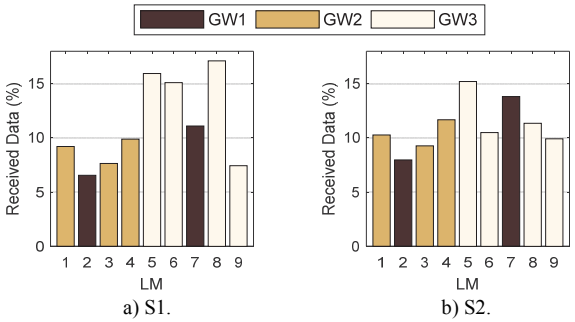
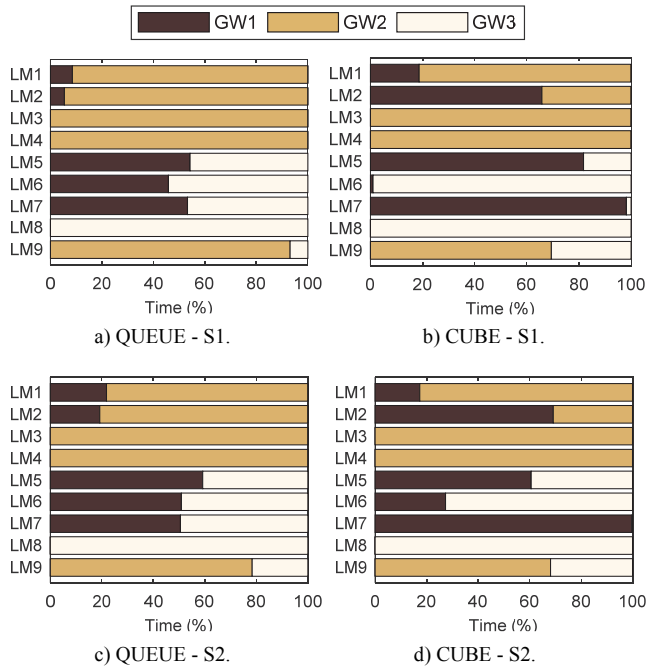


Fig. 9. Percentage of the total data received at each LM.

largest amount of data from the sensor nodes as a result of the concentration of nodes in the working areas.

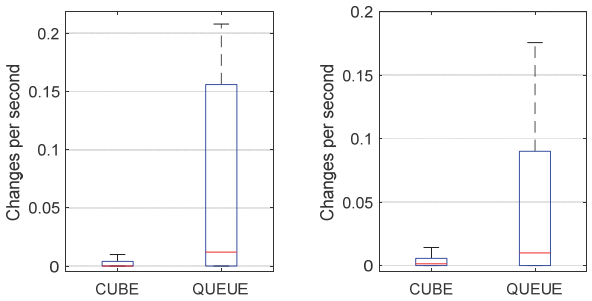
Fig. 8 clearly shows that the fixed assignment of LM to Gateway nodes (fixedGW) results in the largest percentage of lost packets since fixed assignments cannot effectively cope with the spatio-temporal variations of the data. The implementation of load balancing schemes can better cope with such variations, and QUEUE and CUBE considerably reduce the percentage of lost packets in S1 and S2 (Fig. 8). In both scenarios, CUBE outperforms QUEUE. For example, QUEUE reduces the average percentage of lost packets with respect to fixedGW by 69% and 39% in S1 and S2 respectively, whereas CUBE reduces it by 85% and 59%. Different patterns are observed for S1 and S2. In S1, packets are mostly lost in LM5, LM6 and LM8 when fixing the assignment of LMs to Gateways (fixedGW). These LMs receive most of the data transmitted by the sensor nodes (Fig. 9), and they are all connected to Gateway 3 with fixedGW. This overloads the communication channel between the LMs and Gateway 3 which results in the packet losses shown in Fig. 8. CUBE and QUEUE reduce the percentage of lost packets in all LMs compared to fixedGW. Fig. 10 shows the percentage of time that each LM is assigned to each Gateway with QUEUE and CUBE in S1 and S2. Fig. 10 shows that QUEUE and CUBE assign some of the LMs originally attached to Gateway 3 (e.g. LM5 and LM6) to other Gateways in order to balance the load between channels. Fig. 11 shows a box plot of the number of times per second that each LM changes its serving Gateway. An LM changes its serving Gateway when the load balancing scheme estimates that the change is necessary in order to balance the load between the Gateways in the scenario. In Fig. 11, the red line within the box represents the median, and the edges of the box the 25th



a) QUEUE - S1. b) CUBE - S1.

c) QUEUE - S2. d) CUBE - S2.

Fig. 10. Percentage of time that each LM is assigned to a Gateway.



a) S1. b) S2.

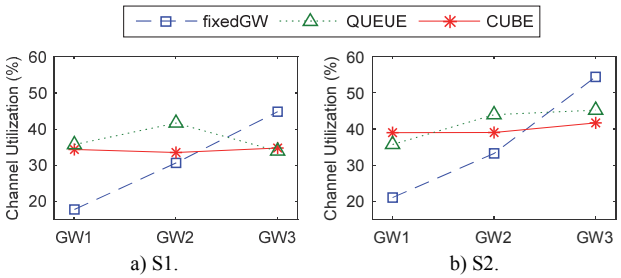
Fig. 11. Box plot of the number of times per second that each LM changes its serving Gateway.

and 75th percentiles. The whiskers represent the minimum and maximum values. Fig. 11 shows that QUEUE results in that 25% of LMs (in particular, LM5, LM6 and LM7) change more than 0.15 times per second their serving Gateway. This is equivalent to changing the serving Gateway every 7 seconds in S1. On the other hand, CUBE demands significantly less changes per second of serving Gateway than QUEUE in S1 (Fig. 11), and hence results in a more stable network operation. For example, with CUBE, approximately 75% of LMs change their serving Gateway every 167 seconds or more in S1. This is equivalent to changing less than 0.005 times per second the serving Gateway. Fig. 10 shows that CUBE assigns LM5 to Gateway 1 during more than 81% of the time, which reduces the load experienced by Gateway 3. This allows LM6 to remain connected to Gateway 3, while considerably reducing the packets lost at LM5, LM6 and LM8 (Fig. 8). It should be noted that CUBE prioritizes the reassignment of LM5 to Gateway 1 (and not LM6) since LM5 is under LOS conditions with Gateway 1 and hence experiences better link quality.

In S2, fixedGW achieves a lower percentage of lost packets in LM7 compared to CUBE and QUEUE (Fig. 8). This is obtained at the expense of significantly increasing the percentage of lost packets at LM5 for fixedGW. This is due to the overload of Gateway 3 since LM5, LM6, LM8, LM9 and the two IP cameras are connected to this Gateway (Fig. 9). LM5 is the LM that receives the largest amount of data from sensors in S2 when mobile sensors concentrate in areas B and C at specific times (Fig. 9.b). So LM5 is also the LM that is mostly affected by the overload of Gateway 3, which explains its packet losses. QUEUE and CUBE are capable of balancing the load between the different Gateways. To this aim, they temporarily assign LM5, LM6 and LM7 to Gateway 1 when the load at Gateway 3 increases (Fig. 10). This significantly reduces the total amount of lost packets (including at LM5). In fact, Fig. 8 shows that QUEUE and CUBE better distribute the packet losses between LMs. The concentration of packet losses in an LM is very negative since LMs receive data sensed within their neighborhood. If an LM loses a large percentage of packets, its serving area risks to be partially disconnected. Although both QUEUE and CUBE reduce the packets lost in the network, they differ in how they assign LMs to Gateways (Fig. 10). For example, QUEUE distributes nearly equally the assignment of LM5, LM6 and LM7 between Gateways 1 and 3 (Fig. 10). On the other hand, CUBE provides more stable assignments to LM6 and LM7, which significantly reduces the changes per second of serving Gateway (Fig. 11). For example, CUBE results in that approximately 75% of LMs change their serving Gateway every 175 seconds or more in S2. This is equivalent to changing less than 0.006 times per second the serving Gateway. On the other hand, 25% of LMs change the serving Gateway every 11 seconds or less in S2.

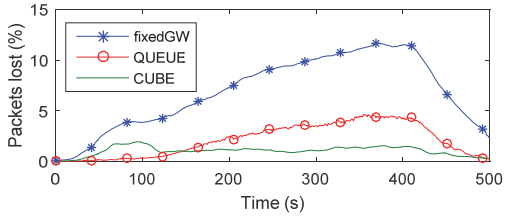
CUBE outperforms QUEUE because it can better balance the load between the Gateways. This is illustrated in Fig. 12 that represents the average CU across all the Gateways. Balancing the load or CU between the three gateways guarantees that none of them will be saturated, and they can hence better support spatio-temporal variations of the data load within the plant.

Fig. 13 represents the evolution of the percentage of packets lost as a function of the time in S1 and S2. The time interval represented in Fig. 13 corresponds to the time during which there are tasks executed in S1 and S2 in the working areas A, B and C. During these time intervals, the network load in these areas increases due to the mobility of nodes and the activation

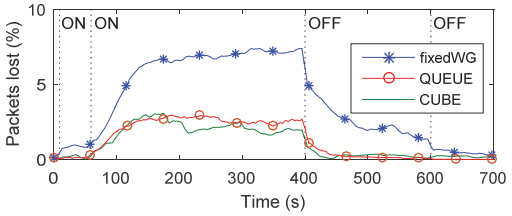


a) S1. b) S2.

Fig. 12. Average Channel Utilization or CU.



a) S1.

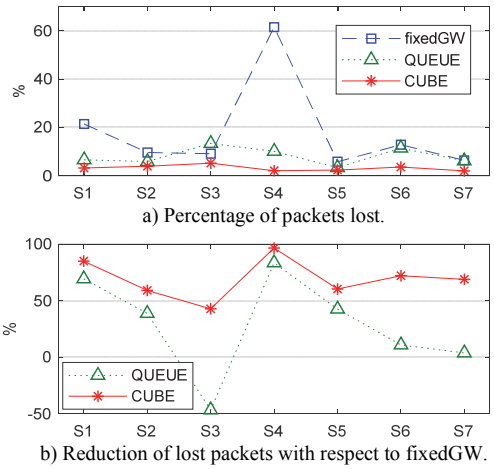


b) S2.

Fig. 13. Percentage of packets lost in S1 and S2.

of IP cameras in S2. The activation and deactivation of these cameras is shown in Fig. 13.b with the ON and OFF marks. Fig. 13.a shows that QUEUE achieves in S1 a slightly lower percentage of packet losses at the start of the time interval, i.e. when the network load has not yet significantly augmented. However, when the load increases (from approximately *t*=100s until *t*=400s), CUBE significantly outperforms QUEUE, and reduces the percentage of lost packets. For example, during the interval [150s, 400s], CUBE reduces the average percentage of lost packets with respect to QUEUE by 64%. In addition, it is important to remember that CUBE significantly reduces the number of times an LM must change its serving Gateway (Fig. 11), and hence guarantees a more stable network operation. In S2, the network load rapidly increases in working areas B and C when the IP cameras are switched on at t=10s and t=60s (Fig. 13.b). In particular, the network starts saturating when the two cameras are active (i.e. after t=60s), and the percentage of lost packets increases. However, CUBE and QUEUE significantly reduce this percentage with respect to the fixed assignment of LMs to Gateways. CUBE is again the scheme that results in the lowest percentage of lost packets and number of changes of serving Gateway (Fig. 11). For example, CUBE reduces the average percentage of lost packets with respect to QUEUE by 17.3% during the interval [150s, 400s] in S2.

Fig. 14 and Fig. 15 compare the performance of CUBE and the reference schemes under different scenarios (defined in Section VII) and evaluation conditions. Fig. 14 shows that CUBE is the most reliable scheme for all the scenarios and evaluations conditions (including when using a different radio propagation model in S3). CUBE always reduces the packets lost in comparison with the fixed deployment (fixedGW) and with QUEUE. Fig. 14 also shows that for certain conditions, CUBE achieves higher gains with respect to QUEUE and fixedGW than those observed in the scenarios S1 and S2. Fig. 15 also shows that in all the scenarios CUBE ensures a more stable network operation than QUEUE since it guarantees fewer changes per second of serving Gateway.



a) Percentage of packets lost.

b) Reduction of lost packets with respect to fixedGW.

Fig. 14. Reliability.

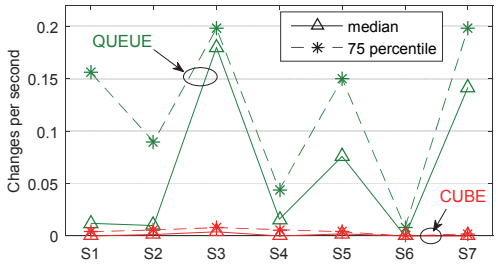


Fig. 15. Median and 75 percentile of the number of times per second that each LM changes its serving Gateway.

IX. Conclusion

This study has presented and evaluated a dynamic load balancing scheme for industrial wireless networks. The scheme has been designed with the objective to support the foreseen spatio-temporal variations of data in IIoT, and support the deployment of reliable and self-organizing industrial wireless networks. The proposed scheme balances the load among nodes taking into account the quality of the wireless links, the amount of data to be transmitted by each node, and the congestion of the wireless channels. All the information needed by the proposed scheme is easily available and measurable at the nodes. The scheme is capable of adapting the configuration of wireless links to the spatio-temporal variations of data in industrial environments, control the signaling overhead, and reduce the number of times that the wireless connections have to be reconfigured.

The conducted study has demonstrated that the proposed load balancing scheme significantly improves the reliability (reduces the packets lost by up to 85%) compared to current deployments where wireless links between nodes are generally predefined and fixed. This is due to the capacity of the proposed scheme to balance the load between the channels, and hence reduce channel saturation as previously illustrated in cellular and wireless networks ([15] and [16]). The proposed scheme also outperforms existing load balancing solutions that base their decision on the data queue length of the wireless nodes. For example, our proposed solution reduces the packets lost in industrial wireless communications

by up to 79% (depending on the scenario) compared to queue-based load balancing schemes. These gains are obtained while also guaranteeing a more stable network operation that significantly reduces the number of times nodes need to reconfigure their wireless links in order to efficiently support spatio-temporal variations of data in industrial environments.

REFERENCES

[1] *Industry 4.0. Digitalisation for productivity and growth*, European Parliamentary Research Service, Sept. 2015.
[2] R. Drath, A. Horch, "Industrie 4.0: Hit or Hype?", *IEEE Industrial Electronics Magazine*, vol.8, no. 2, pp. 56-58, June 2014.
[3] Plattform Industrie 4.0, "Network-based communication for Industrie 4.0", *Publications of Plattform Industrie 4.0*, April 2016. Available at http://www.plattform-i40.de/I40/Redaktion/EN/Downloads/Publikation/network-based-communication-for-i40.html. Last access on 2018/08/02.
[4] C. Lu et al., "Real-Time Wireless Sensor-Actuator Networks for Industrial Cyber-Physical Systems", *Proc. of the IEEE*, vol. 104, no. 5, pp. 1013-1024, May 2016.
[5] M. Al-Rousan and D. Kullab, "Real-Time Communications for Wireless Sensor Networks: A Two-Tiered Architecture", *International Journal of Distributed Sensor Networks*, vol. 5, no. 6, pp. 806–823, 2009.
[6] X. Li, et al, "A review of industrial wireless networks in the context of Industry 4.0", *Wireless Networks*, vol. 23, no. 1, pp. 23-41, Jan. 2017.
[7] J.R. Gisbert, et al., "Integrated system for control and monitoring industrial wireless networks for labor risk prevention", *Journal of Network and Computer Applications*, vol. 39, pp. 233-252, March 2014.
[8] M.C. Lucas-Estañ, et al., "A Software Defined Hierarchical Communication and Data Management Architecture for Industry 4.0", *Proc. of IEEE/IFIP WONS* 2018, Isola 2000, France, 6-8 Feb. 2018.
[9] M. Domingo-Prieto, T. Chang, X. Vilajosana, T. Watteyne, "Distributed PID-Based Scheduling for 6TiSCH Networks", *IEEE Communications Letters*, vol. 20, no. 5, pp. 1006-1009, May 2016.
[10] S. Montero, et al., "Impact of Mobility on the Management and Performance of WirelessHART Industrial Communications", *Proc. 17th IEEE ETFA Conference*, Kraków (Poland), 17-21 Sept. 2012.
[11] L. Seno, et al., "Enhancing Communication Determinism in Wi-Fi Networks for Soft Real-Time Industrial Applications", *IEEE Trans. on Industrial Informatics*, vol. 13, no. 2, pp. 866-876, April 2017.
[12] F. Tramarin, et al., "Improved Rate Adaptation strategies for real-time industrial IEEE 802.11n WLANs", *Proc. 20th IEEE ETFA Conference*, Luxembourg, pp. 1-8, 8-11 Sept. 2015.
[13] J. Gozalvez, M. Sepulcre, J.A. Palazón, "On the feasibility to deploy mobile industrial applications using wireless communications", *Computers in Industry*, vol. 65, no. 8, pp. 1136-1146, Oct. 2014.
[14] B. Holfeld, et al., "Wireless Communication for Factory Automation: an opportunity for LTE and 5G systems", *IEEE Communications Magazine*, vol. 54, no. 6, pp. 36-43, June 2016.
[15] C. Ran, S. Wang, C. Wang, "Balancing backhaul load in heterogeneous cloud radio access networks", *IEEE Wireless Communications*, vol. 22, no. 3, pp. 42-48, June 2015.
[16] Q. Ye, et al. "User Association for Load Balancing in Heterogeneous Cellular Networks", *IEEE Trans. on Wireless Communications*, vol. 12, no. 6, pp. 2706-2716, June 2013.
[17] M. Collotta, et al., "Dynamic load balancing techniques for flexible wireless industrial networks", *Proc. IEEE IECON 2010*, pp. 1329-1334, Glendale, USA, 2010.
[18] U. Ashraf, S. Abdellatif, G. Juanole "Gateway Selection in Backbone Wireless Mesh Network", *Proc. WCNC*, pp. 1-6, Budapest, Hungary, 2009.
[19] C. Basaran, et al., "Hop-by-hop congestion control and load balancing in wireless sensor networks", *Proc. IEEE 35th LCN*, pp. 448-455, Denver, USA, 2010.
[20] M.C. Lucas-Estañ, et al., "An Experimental Evaluation of Redundancy in Industrial Wireless Communications", in *Proc. 23rd IEEE ETFA Conference*, Turin, Italy, 4-7 September, 2018.
[21] J. He, W. Guan, L. Bai, K. Chen, "Theoretic Analysis of IEEE 802.11 Rate Adaptation Algorithm SampleRate", *IEEE Communications Letters*, vol. 15, no. 5, pp. 524-526, May 2011.
[22] IEEE Standard for Information technology--Telecommunications and information exchange between systems Local and metropolitan area networks--Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, 7 Dec.2016.
[23] IBM ILOG CPLEX Optimizer, URL: http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/. Last access on 2018/08/02.
[24] E. Tanghe et al., "The industrial indoor channel: large-scale and temporal fading at 900, 2400, and 5200 MHz", *IEEE Trans. on Wireless Communications*, vol. 7, no. 7, pp. 2740-2751, July 2008.
[25] 3GPP; Technical Specification Group Services and System Aspects; Study on Communication for Automation in Vertical Domains (Release 16), 3GPP TR 22.804 v16.1.0, Sept. 2018.

M.Carmen Lucas-Estañ received a Telecommunications Engineering degree in February 2007 and a Ph.D. in Industrial and Telecommunications Technologies in July 2012, both from Universidad Miguel Hernández de Elche (UMH), Spain. Since 2007, she is with UMH, where she is currently a part-time professor and a post-doc researcher at the UWICORE research laboratory. She received an Orange Foundation Award at the 2007 Spanish Young Scientist Contest organized by the Spanish Ministry of Education. She is currently working on industrial wireless networks, radio resource management, and D2D communications. She was Publicity Co-Chair in IWSS 2017 and IEEE VTC2018-Fall, and track-chair in IEEE VTC2015-Spring.

Javier Gozalvez received an electronics engineering degree from the Engineering School ENSEIRB (Bordeaux, France), and a PhD in mobile communications from the University of Strathclyde, Glasgow, U.K. Since October 2002, he is with the Universidad Miguel Hernández de Elche (UMH), Spain, where he is currently a Full Professor and Director of the UWICORE laboratory. At UWICORE, he leads research activities in the areas of vehicular networks, multi-hop cellular networks and D2D communications, and wireless industrial networks. He is an elected member to the Board of Governors of the IEEE Vehicular Technology Society (VTS) since 2011, and served as President of the IEEE VTS in 2016 and 2017. He was an IEEE Distinguished Lecturer for the IEEE VTS, and currently serves as IEEE Distinguished Speaker. He is the Editor in Chief of the IEEE Vehicular Technology Magazine. He was the General Co-Chair and founder of the IEEE Connected and Automated Vehicles Symposium 2018, and the Co-Chair of the IEEE VTC-Spring 2015 conference in Glasgow (UK), ACM VANET 2013, ACM VANET 2012 and 3rd ISWCS 2006.