

Neighbor Discovery for Industrial Wireless Sensor Networks with Mobile Nodes

Sergio Montero, Javier Gozalvez, Miguel Sepulcre
UWICORE, Ubiquitous Wireless Communications Research Laboratory
Universidad Miguel Hernández de Elche (UMH), Elche (Alicante), Spain
smontero@umh.es, j.gozalvez@umh.es, msepulcre@umh.es

Abstract— Industrial wireless sensor networks can facilitate the deployment of a wide range of novel industrial applications, including mobile applications that connect mobile robots, vehicles, goods and workers to industrial networks. Current industrial wireless sensor standards have been mainly designed for static deployments, and their performance significantly degrades when introducing mobile devices. One of the major reasons for such degradation is the neighbor discovery process. This paper presents and evaluates two novel neighbor discovery protocols that improve the capability of mobile devices to remain connected to the industrial wireless sensor networks as they move. The proposed protocols exploit topology information and the nature of devices (static or mobile) to reliably and rapidly discover neighbor devices. This is achieved in some cases at the expense of increasing the number of radio resources utilized and the energy consumed in the discovery process. The proposed solutions have been designed and evaluated considering the WirelessHART standard given its widespread industrial adoption. However, they can also be adapted for the ISA100.11a and IEEE 802.15.4e standards.

Keywords Industrial wireless sensor networks, IWSN, WirelessHART, ISA100.11a, IEEE 802.15.4e, mobility, neighbor discovery, factories of the future, industry 4.0, industrial wireless communications, industrial wireless networks.

1. Introduction

Industrial Wireless Sensor Networks (IWSNs) can help reduce the cost and time needed for the installation and maintenance of cables and machinery, enhance the flexibility and reconfigurability of a factory, and facilitate the introduction of healthcare solutions [1][2]. Current IWSNs mainly focus on static deployments and devices, but there is a growing interest in utilizing IWSNs for connecting mobile subsystems or devices. For example, NAMUR (User Association of Automation Technology in Process Industries) has established a "Mobile Automation" working group (WG 4.15) to study fields of application for wireless technologies and mobile applications in process automation.

Current IWSN standards include WirelessHART [3], ISA100.11a [4] and IEEE 802.15.4e [5] for industrial automation and control applications. Despite their differences, both standards share some fundamental wireless technologies and mechanisms [6], e.g. a centralized network management to provide the reliability and latency levels required by industrial applications. IWSNs still face significant challenges to ubiquitously guarantee the reliability and latency requirements of industrial applications, in particular when considering mobile nodes (robots, vehicles, goods, people, etc.). In fact, the mechanisms defined in WirelessHART for joining, discovering, scheduling or routing are currently not optimized for scenarios where mobile devices would require permanent network connectivity. So, even if WirelessHART considers the use of handheld devices, these handheld devices can only communicate with the attached device and cannot maintain network connectivity as they move. In this context, studies such as [7] and [8] have highlighted the need to design new mechanisms that reduce the time required to discover neighbor devices and the time to reconfigure the network before mobile nodes can be integrated in existing WirelessHART networks. Mobility management mechanisms are also necessary for mobile devices to remain connected to the network [9]. Such connectivity is also significantly influenced by the Neighbor Discovery Protocols (NDP) [10]. Devices utilize the NDPs to discover their one-hop neighbors. Adequate NDPs are necessary to reduce the probability of mobile devices to lose network connectivity and minimize the time to discover neighbors in case the connectivity is temporarily lost. It is important to highlight that the discovery process can only be executed by devices that have previously joined the network. As a result, the NDPs are independent of the process followed by devices to join the network.

NDPs can be classified as probabilistic or deterministic [11]. Deterministic NDP schemes ensure a bounded discovery time. This is generally done by transmitting discovery packets on dedicated radio resources in order to avoid packet collisions. On the other hand, probabilistic NDP schemes are simpler to implement and require fewer radio resources and lower energy consumption for the discovery process. The NDP scheme included in WirelessHART is probabilistic, and cannot hence guarantee strict upper bounds on the discovery latency. This might not be critical when nodes are static and maintain the same neighbors for long periods of time. However, the capacity to rapidly detect neighbors is critical when considering mobile devices. In this context, this paper proposes two novel NDP schemes that improve the neighbor discovery process under the presence of mobile nodes in IWSNs. The first proposal is a deterministic NDP scheme that exploits information about the network topology that is exchanged among devices.

The second proposal combines the advantages of deterministic and probabilistic NDP policies. It does so by applying probabilistic policies for static devices, and deterministic ones for mobile devices. This approach helps reducing the discovery overhead while providing the necessary capacity for mobile nodes to detect neighbors. This study is conducted under the framework of the WirelessHART industrial standard, and the proposed NDP schemes do not affect other existing WirelessHART mechanisms such as the process followed by devices to join the network or the routing [3]. The obtained results demonstrate that the proposed schemes significantly improve the capacity of mobile devices to discover neighbors and the time needed to detect them compared to the existing WirelessHART NDP solution. These benefits are achieved, for some of the proposed NDP protocols, at the expense of increasing the number of radio resources utilized and the energy consumed in the discovery process compared to the current WirelessHART NDP process.

The paper is organized as follows. Section 2 reviews the state of the art on neighbor discovery protocols in IWSNs. Section 3 describes the WirelessHART standard and its NDP scheme. Section 4 describes the proposed NDP schemes, and Section 5 defines the metrics utilized to evaluate their performance. Section 6 presents analytical performance models for the NDP schemes under evaluation, and Section 7 compares their performance through simulations. The main contributions and conclusions of this study are summarized in Section 8.

2. State of The Art

Managing the mobility of nodes usually requires handover mechanisms that implement three different phases: information gathering, decision and execution. The information gathering phase is responsible for monitoring and collecting all context information. Based on the collected information, a handover decision would then be taken, including the selection of the new connecting node. Executing a handover can require, for example, changing the assigned frequency channel or time slot. NDP schemes are part of the information gathering phase, and are the focus of this study.

The study in [12] presents an excellent review of existing NDP schemes in wireless networks. All NDPs generally fall into two categories: probabilistic or deterministic [11]. In deterministic NDPs, nodes transmit discovery packets following a predefined schedule that guarantees a bounded discovery time [13]. Probabilistic NDPs cannot provide such guarantee since nodes transmit their discovery packets at randomly chosen times and packet collisions can be produced. On the other hand, probabilistic NDPs are simpler to implement and utilize fewer radio resources and lower energy consumption for the discovery process. The WirelessHART NDP scheme and the birthday protocol [14] are examples of probabilistic NDPs implemented over Time Division Multiple Access (TDMA). With the birthday protocol [14], nodes can choose for each time slot dedicated to the discovery process whether they are in transmission, reception or sleep mode following predefined probabilities. WirelessHART does not employ the sleep mode, and each device goes into transmission mode after a random waiting time. To discover neighbors, WirelessHART requires devices to be in reception mode in all discovery slots in which they are not transmitting.

The time necessary to discover neighbors can be arbitrarily long in the case of probabilistic NDPs. Probabilistic NDPs might hence not be suitable for mobile networks that require a strict upper bound on discovery latency [15]. Deterministic NDP schemes are hence usually utilized by mobile devices for neighbor discovery [15]. For example, cellular systems use deterministic schemes to identify the base station to which a node should connect. Cellular networks periodically inform mobile terminals about the neighboring cells, and the mobiles perform radio measurements to identify whether they should change their serving base station [16]. Several deterministic NDPs explicitly or implicitly use brute force techniques to ensure the detection of neighbors before a given deadline [17]. In this case, nodes try to remain in reception mode as long as possible if they do not have any other packet to transmit. NDP schemes that use brute force are not very energy efficient. To reduce energy consumption, schemes such as those presented in [18], [19] and [20] limit the number of time slots under which a node is in reception mode to those where it is known that neighbor nodes can transmit discovery packets. In this case, the challenge is how to identify and inform each node about the time slots in which it has to be in reception mode to detect neighbors. The NDP scheme proposed in [18] predicts future contacts statistically using network science in order to reduce the energy consumption. The NDP scheme proposed in [19] uses reinforcement learning techniques to dynamically identify the time slots under which a node should be in reception mode. For improving the learning process, and to account for pattern variations, the NDP scheme in [19] also allows nodes to monitor other time slots.

Neighbor discovery is particularly challenging when devices can operate over multiple channels like it is the case of WirelessHART. Multi-channel neighbor discovery policies are also necessary when applying deterministic NDP schemes to standards that allow for the simultaneous use of several channels. The multi-channel NDP proposed in [21] operates with heterogeneous duty cycles and without clock synchronization. Another example is the McDisc proposal [22] that utilizes multiple available channels to establish a multi-channel neighbor discovery schedule that reduces energy consumption and increases the discovery reliability. A different approach is considered in [20] where a central entity informs each node about the time slots under which it should be in reception mode to detect neighbors. The work proposed in [20] is the first study that considered the neighbor discovery challenge in WirelessHART networks that include mobile devices. Other relevant NDP schemes have been reported in

[10], [13] and [17]. However, they cannot be directly applied in WirelessHART since [10] considers full-duplex technology, [13] divides each slot into two subslots, and [17] assumes that if two devices are awake during the same slots they can mutually discover each other.

3. WirelessHART

WirelessHART defines a centralized network architecture. It is based on the IEEE 802.15.4 physical layer and operates in the 2.4GHz ISM band. Although 16 channels are available in this band, only channels 11-25 can be used by WirelessHART devices. On top of the IEEE 802.15.4 physical layer, WirelessHART adds a TDMA scheme combined with Frequency Hopping for improved robustness and capacity. A communications link is therefore defined in WirelessHART by a time slot and a frequency channel. WirelessHART allows transmitting in up to 15 different frequency channels, but it generally does not allow instantaneously reusing one slot of a given frequency channel for more than one transmission. Time slots are 10ms long, and are organized in superframes that are managed by the *Network Manager*. WirelessHART defines different types of superframes. *Data superframes* are used to schedule links for data transmissions. The *Management Superframe* is used to schedule all links related to the network management. The *Management Superframe* has a periodicity T_N of 64 seconds (equivalent to 6400 slots). The *Network Manager* organizes slots into superframes, and is responsible among other functions, for allocating links to network devices in order to transmit Data Link Protocol Data Units (DLPDUs). WirelessHART defines five types of DLPDU:

- *Data* is used to transmit data to a final destination device.
- *ACK* is the immediate link level response to the reception of a *Data* from a source device.
- *Disconnect* is used by a device to inform its neighbor devices that it is leaving the network.
- *Advertise* is used by network devices to broadcast network information in dedicated links to devices trying to join the network. This information is used by new devices in order to initiate the join process.
- *Keep-alive* is used for neighbor discovery and connectivity maintenance. When a device receives a *Keep-alive*, it updates its neighbor list. *Keep-alive* are transmitted by devices in the so called *Discovery Links*, which are links shared by all devices in the network.

ACK, *Disconnect*, *Advertise* and *Keep-Alive* DLPDUs are generated and consumed at the data-link layer [3]. These DLPDUs are not forwarded to the network layer.

The *Network Manager* manages the WirelessHART network, and in particular the neighbor discovery process and the join process followed by devices to access the network. This paper focuses on the neighbor discovery process. However, the join process is also described below because some of the proposed NDP schemes make use of the *Advertise* sent by devices to access the network (join process).

A. WirelessHART join process

The process followed by new WirelessHART devices to access the network utilizes the transmission and reception of *Advertise* on dedicated links. The *Network Manager* schedules all devices that have joined the network to transmit one *Advertise* per *Management Superframe*. A new device that wants to join the network must first be in reception mode to receive at least an *Advertise* from another network device. The new device will only accept the advertised information if it includes the device's pre-configured network ID. After the reception of an *Advertise*, the new device that wants to join the network will keep listening for an additional amount of time trying to find more neighbors¹. If during this time the new device receives multiple *Advertise* (sent by different devices that are already part of the network), it must select the best candidate device to send it a *Join Request* message in order to access the network. In particular, it selects as best candidate the device from which the *Advertise* was received with the highest *signal strength* and *join priority*.

The *Join Request* is sent using a pre-programmed join key. A different join key is used by each device to authenticate itself and encrypt the data during its process to access the network. Multiple new devices might compete to join the network at the same time through the best candidate device, which could generate collisions when sending the *Join Request* messages. To reduce this probability, new devices must wait a random time before sending their *Join Request* that is acknowledged at the data link layer. If the *Join Request* is not successfully received and acknowledged by the selected best candidate device, the new device will send it again after a random period of time. Once it is successfully received by the selected best candidate device, the *Join Request* is forwarded to the *Network Manager*. The *Network Manager* decides whether the new device should have access to the network or not (for example if the join key is not correct). If the decision is positive, the *Network Manager* sends a *Join Reply* to the new device that includes authorization data and the scheduling of its time slots. This new schedule may also involve the allocation or reallocation of slots to other nodes in order to establish new data and management multi-path routes.

¹ This is the case because the reliability of a WirelessHART network largely derives from the fact that every device has multiple neighbors, and multiple routes can hence be established between any pair of nodes.

B. WirelessHART neighbor discovery process

All devices that have joined the network use the neighbor discovery process to detect each other. Every time a device discovers a neighbor device, it updates its neighbor table and periodically informs the *Network Manager* so that the routing tables and forwarding routes can be updated. An adequate operation of the network therefore requires the capacity to constantly detect or discover neighbor devices. This can be particularly challenging in the case of mobile devices as the neighborhood relations constantly change when devices move [7]. If neighbor devices cannot detect each other in time, they can get disconnected from the network. In this case, they will need to perform again a join process to regain connectivity.

The WirelessHART neighbor discovery protocol (referred to as WH in this study) is based on the transmission and reception of *Keep-alive* sent by neighbor devices on a shared *Discovery Link*. The *Network Manager* includes at least one *Discovery Link* in each *Management Superframe* where each network device randomly sends a *Keep-alive* or listens to the possible transmission of a *Keep-alive* from one of its neighbor devices. To transmit a *Keep-alive*, a device first selects a random waiting time between 0 and *Discovery_time*. When this time expires, the device transmits a *Keep-alive* at the first available *Discovery Link*, and selects a new random waiting time to transmit the next *Keep-alive*. In order to receive *Keep-alive* messages sent by neighbor nodes, the devices must be in reception mode in all *Discovery Links* in which they are not transmitting their own *Keep-alive*. If a device receives a *Keep-alive* from a device that was not previously identified as a neighbor, it stores it in its neighbor list and informs the *Network Manager*.

4. Neighbor Discovery Proposals

The work proposed in [20] was the first to address the neighbor discovery challenge when considering mobile nodes in WirelessHART networks. This study therefore considers the proposal reported in [20] and the NDP protocol defined in WirelessHART (WH) as benchmark schemes.

A. Listen Advertise Network

The authors proposed in [20] the LAN (Listen Advertise Network) neighbor discovery protocol in order to improve the capacity of mobile nodes to detect neighbor devices and reduce the time needed for their detection. The WH protocol listens for *Keep-alive* on shared *Discovery Links* to detect neighbor devices. Instead, LAN is a deterministic NDP that proposes that each device listens for *Advertise* transmitted on dedicated links by all the other network devices that are already part of the network. As detailed in section 3.A, *Advertise* are used by new devices to get access to the network. LAN uses the *Advertise* messages for the neighbor discovery process. However, it does not change the process defined in WirelessHART for new devices to join the network, and does not modify the content of the *Advertise* or their transmission timing. Since *Advertise* messages are transmitted more frequently than *Keep-alive* messages, LAN can significantly reduce the average time needed to detect neighbor devices compared to WH [20]. This is the case because one *Advertise* is transmitted by each device in a dedicated link in each *Management Superframe*. On the other hand, *Keep-alive* messages are transmitted in links that are shared by all devices. LAN requires each device to be in reception mode in the links where the rest of network devices transmit their *Advertise* message. The information about when all *Advertise Links* are scheduled is sent by the *Network Manager* to all network devices [20].

B. Listen for Close Advertises

LAN can easily be implemented in WirelessHART, and it reduces the time needed to detect a neighbor device compared to WH. However, LAN does not maximize the utilization of radio resources nor minimize the energy consumption since all devices need to be in reception mode whenever a device transmits an *Advertise*. This can be particularly inefficient when the device transmitting the *Advertise* is distant, and therefore the probability to receive the *Advertise* is low. To overcome this inefficiency, this paper proposes the LCA (Listen for Close Advertises) protocol that can be integrated into WirelessHART. With LCA, each device only tries to receive *Advertise* messages from its one- and two-hop neighbors. To identify two-hop neighbors, LCA requires each device to transmit all its one-hop neighbors in its *Advertise*. The number of neighbors that can be listed in each *Advertise* is limited by the number of bytes that can be transmitted in an *Advertise Link*. Several *Advertise* might then be necessary to transmit the complete list of one-hop neighbors. When a device receives an *Advertise*, it includes in its list of two-hop neighbors those devices that are listed in the received *Advertise* and are not identified as one of its one-hop neighbors. A device is maintained in the neighbor table until a timeout expires without receiving (directly or through other nodes) any update about this device. LCA requires devices to only try to receive *Advertise* from their one-hop or two-hop neighbors. As a result, LCA improves the utilization of the radio resources and energy consumption compared to LAN since it significantly reduces the time a device needs to be in reception mode to listen for *Advertise*. A device implementing LCA can utilize this free time to perform other functions, including transmitting data.

C. Listen Advertise Network for mobile devices

LAN and LCA seek reducing the time necessary to detect neighbor devices in WirelessHART. This is critical for mobile devices that constantly change their neighbor relations, and can therefore get disconnected from the network if they do not detect neighbor

devices in time. On the other hand, static devices do not frequently change their neighbors, and therefore do not require short neighbor detection times. LAN and LCA do not differentiate between static and mobile devices, and treat them equally in terms of the neighbor discovery process. In this context, this paper proposes an additional NDP protocol, referred to as LANm (Listen Advertise Network for mobile devices), that exploits the different needs and characteristics of mobile and static devices. A device is considered to be mobile if it has the ability to move, regardless of whether it is in constant movement or can be partially static; e.g. a device attached to a forklift is considered mobile, while a device attached to a tufting machine is considered static. This characteristic (static or mobile) is therefore inherent to each device and does not change over time.

LANm applies WH to all (static and mobile) devices that will then listen for *Keep-alive* sent on the shared *Discovery Link* in order to detect neighbor devices. In addition, LANm applies the LAN scheme to mobile devices². In this case, the *Network Manager* only schedules mobile devices to be in reception mode when all network devices (mobile and static) send their *Advertise*. To inform the *Network Manager* about whether a given device is mobile or static, one of the bits currently reserved for future use in the *specifier* field of the *Join Request* DLPDU could be used [3].

By combining WH and LAN, LANm seeks to improve the utilization of the radio resources and energy consumption while guaranteeing the capacity to quickly detect neighbor devices. This is the case since WH consumes a very small number of radio resources, and LAN improves the capacity to detect neighbor devices. LANm is the first scheme that combines deterministic and probabilistic NDP approaches in WirelessHART.

5. Metrics

The performance of the NDP schemes is evaluated using different metrics. A key metric is the discovery probability. The discovery probability is defined as the probability that a device i discovers another device j or vice versa (P_{disc}), considering that they are under each other's coverage range only during a time period t_{cov} . Another important metric is the average time required to detect a neighbor device (t_{det}). This metric is estimated as the average time elapsed from the moment a device enters the coverage range of another device until the moment it is detected.

A mobile device does not require detecting all its neighbor devices to maintain the connection to the network. Such connection can be maintained if the mobile device is always connected to at least one neighbor device that is connected to the network. A relevant metric is then the probability that a mobile device remains connected to the network (P_{conn}) while it moves around its coverage area. P_{conn} is here computed as the ratio between the number of evaluations during which the mobile device did not lose connectivity to the network and the number of conducted evaluations for a particular scenario.

Several metrics are utilized to analyze the efficiency of NDPs. One of these metrics is the average *Number of Bytes Consumed* (NBC) by the discovery process per *Management Superframe*. NBC takes into account the bytes needed for the correct operation of the NDP, and the bytes that are left unused as certain links that are not directly used by the NDP protocol are blocked and cannot be utilized for other purposes either. This is for example the case when all devices must be in reception mode in one link, and no device can actually transmit/receive data in a different channel. The efficiency of NDPs is also analyzed by means of the total number of links used for neighbor discovery per *Management Superframe*. Differently from NBC, this metric does not take into account whether the dedicated links are utilized or not.

Finally, an important aspect when dealing with battery-powered mobile devices is the energy consumption. The energy efficiency is here investigated by estimating the average energy consumed (E_d) by the discovery process of each NDP per device and *Management Superframe*. This metric is estimated considering the energy consumed by the transmission or reception of the necessary DLPDUs, the energy consumed when devices are in idle or reception mode, and the energy consumed by the configuration of devices (E_c) following the WirelessHART standard.

6. Analytical Performance Modeling

This section presents analytical performance models for the different NDP schemes. In particular, it presents analytical expressions of the neighbor discovery probability, the average time needed to detect a neighbor, the efficiency and the energy consumption³. These expressions are derived to help understand the influence of different parameters, and provide a tool to obtain first estimates of the performance characterizing each NDP scheme.

² It is important noting that applying LAN to mobile devices instead of LCA minimizes the impact on static devices. If LCA was applied to mobile devices, static devices would need to modify their *Advertise* in order to include their list of neighbors. With LCA, a mobile node could only detect a device i if it had previously detected a device j that had i in its list of neighbors. In any case, tests were also done applying LCA to mobile devices, and no significant differences (in performance and energy consumption) were observed compared to when mobile devices implement LAN.

³ A first version of some of these expressions was presented in [20] for LAN and WH using a simplified propagation model. The expressions presented in this section have been derived considering more realistic propagation and interference conditions.

A. System model

This study considers an industrial wireless network with H devices. All H devices have previously joined the network and there are not new devices trying to join the network. A device can be either a static device or a mobile device. As previously mentioned, a device is considered to be mobile if it has the ability to move, regardless of whether it is in constant movement or is partially static. Static devices support the connection of mobile devices to the network. The number of mobile devices in the network is M .

The *Network Manager* is in charge of the scheduling function for the neighbor discovery process on the *Management Superframe*. A *Management Superframe* has N_s slots and the duration of each slot is t_s ; the duration of a *Management Superframe* is then $T_N = N_s t_s$. In each slot, ρ bytes can be transmitted, and the number of frequency channels is defined as C .

The WirelessHART standard provides the *Network Manager* some flexibility for the scheduling of *Advertise Links* and *Discovery Links* since it does not define specific scheduling algorithms. Without loss of generality, this study considers that each device is allocated an *Advertise Link* per *Management Superframe* to transmit its *Advertise*. This means that H *Advertise* are transmitted per *Management Superframe*. Also, this study considers that there is one *Discovery Link* in every *Management Superframe* which is shared by all devices. Therefore the *Discovery Link* has a periodicity of T_N seconds.

The mathematical notation used employs subscripts to refer to nodes or protocols. Superscripts are used to refer to the number of deployed nodes (H) or the number of *Management Superframes* under evaluation (S). Table 1 lists the important symbols and parameters defined in this section to support the readability of the proposed analytical performance models.

Table 1 - Symbols and parameters

B	Length of a <i>Keep-alive</i> message in bytes
B_A	Length of the Device ID in bytes
B_D	Number of bytes transmitted by the <i>Network Manager</i> to schedule an <i>Advertise Link</i>
C	Number of frequency channels
DL	Number of <i>Discovery Links</i> in a <i>Management Superframe</i>
DL_{dl}	Link within the <i>Management Superframe</i> for the dl <i>Discovery Link</i>
DLN_i	Number of <i>Management Superframes</i> within <i>Discovery_time</i> for device i
E_c	Energy consumed by the configuration of devices
E_{idle}	Average energy consumed per device in idle mode
E_{list}	Energy consumed by a device when transmitting the neighbor list
E_{Rx-KA}	Energy consumed by a device when receiving a <i>Keep-alive</i>
E_{Rx-Ad}	Energy consumed by a device when receiving an <i>Advertise</i>
E_{Tx-KA}	Energy consumed by a device when transmitting a <i>Keep-alive</i>
E_{Tx-Ad}	Energy consumed by a device when transmitting an <i>Advertise</i>
H	Number of deployed nodes
H_{h1}	Number of one-hop neighbors of device h
H_{h2}	Number of one- or two-hop neighbors of device h
hop_h	Number of hops between device h and the <i>Network Manager</i>
M	Number of mobile devices
N	Noise power level
N_s	Number of slots within a superframe
P_i	Probability that a device i transmits a <i>Keep-alive</i> in one <i>Discovery Link</i>
P_i^{dl}	Probability that device i transmits in the dl <i>Discovery Link</i> of a <i>Management Superframe</i>
$P_{i,j}^H$	Probability that a device i detects device j in one <i>Discovery Link</i> or <i>Advertise Link</i>
$P_{i,j}^{H,S}$	Probability that a device i receives at least one <i>Advertise</i> or one <i>Keep-alive</i> in S <i>Discovery</i> or <i>Advertise Links</i>
$P_{i,j}^S$	Probability that devices i and j are under each other's coverage range during S <i>Advertise/Discovery Links</i>
$P_{i \leftrightarrow j}^S$	Probability that devices i and j discover each other at least once in S <i>Management Superframes</i>
P_{i-NRx}	Probability that the received signal strength of a <i>Keep-alive</i> is below the minimum signal needed to start the reception
$P_{i-Rx-Ad(j)}$	Probability that device i receives an <i>Advertise</i> from j with a signal strength above the minimum signal needed to start the reception
$P_{rec}(i,h)$	Received signal power level at device i from the interfering device h that simultaneously transmits a packet
$P_{rec}(i,j)$	Received signal power level at device i from device j
Q_{ij}	Number of hops between i and j
R_i	Probability that a device i is in reception mode in one <i>Discovery Link</i>
S	Number of <i>Management Superframes</i> under evaluation
$SINR_{i,j}^H$	SINR experienced by device i when receiving a packet from device j under the presence of H devices
t	Elapsed time under evaluation
t_s	Time slot duration
t_{cov}	Time in which the PDR between i and j is higher than PDR_{min}
T_N	Periodicity of the <i>Management Superframe</i>
ρ	Number of bytes that can be transmitted in a link

B. Discovery probability

1) WirelessHART Neighbor Discovery

The WirelessHART neighbor discovery process is based on the transmission and reception of *Keep-alive* in *Discovery Links*. In the most general case where there are DL *Discovery Links* in a *Management Superframe*, the probability of a given device i to be in transmission or reception mode is different for each of the DL *Discovery Links*. The probability that device i transmits in the dl *Discovery Link* ($1 \leq dl \leq DL$) of a *Management Superframe* (P_i^{dl}) can be expressed as:

$$P_i^{dl} = \frac{\frac{DL_{dl} - DL_{dl-1}}{N_S}}{\frac{DLN_i - 1}{2} + \sum_{x=1}^{dl} \left(\frac{DL_x - DL_{x-1}}{N_S} \sum_{y=x}^{DL} \frac{DL_y - DL_{y-1}}{N_S} \right)} \quad (1)$$

where $DL_{dl} - DL_{dl-1}$ is the number of links between the dl *Discovery Link* and the previous *Discovery Link*, $DLN_i = \lceil \text{Discovery_time} / T_N \rceil$ is the number of *Management Superframes* within *Discovery_time* for device i , and N_S is the number of slots in a *Management Superframe*.

The *Network Manager* has certain flexibility to schedule the *Discovery Links*, and it is reasonable to uniformly distribute them over the *Management Superframe* to improve the use of available radio resources. For example, if two *Discovery Links* in a *Management Superframe* were scheduled in consecutive slots, the second one will almost never be used to transmit a *Keep-alive*. This study therefore assumes that *Discovery Links* are uniformly distributed in the *Management Superframe*, and without loss of generality, we assume that there is only one⁴ *Discovery Link* in every *Management Superframe*, i.e. $DL=1$. In this case, $DL_{dl} - DL_{dl-1} = N_S$ and the expression of the probability that device i transmits in the *Discovery Link* is simplified as follows:

$$P_i^{dl} = P_i = \frac{1}{\frac{DLN_i - 1}{2} + 1} = \frac{2}{DLN_i + 1} \quad (2)$$

The probability that a device i is in reception mode in one *Discovery Link* (R_i) can be expressed as:

$$R_i = 1 - P_i = \frac{DLN_i - 1}{DLN_i + 1} \quad (3)$$

Device i will discover device j if device j transmits a *Keep-alive*, device i is in reception mode, and the Signal to Interference and Noise Ratio (SINR) is sufficient to correctly receive the *Keep-alive*. Let's consider that H devices can interfere the transmission from device j . In this case, the probability that device i discovers device j in one *Discovery Link* can be expressed for WH as:

$$P_{i,j}^H |_{WH} = P_j \cdot R_i \cdot PDR(\text{SINR}_{i,j}^H) \quad (4)$$

where $\text{SINR}_{i,j}^H$ is the SINR experienced by device i when receiving a *Keep-alive* from device j under the presence of H neighbors, and $PDR(\text{SINR}_{i,j}^H)$ is the Packet Delivery Ratio (PDR) at device i when experiencing $\text{SINR}_{i,j}^H$. If we assume that a single bit error out of B bytes is enough to corrupt a *Keep-alive*, it is possible to express the PDR as a function of the Bit Error Rate (BER):

$$PDR(\text{SINR}_{i,j}^H) = \left(1 - \text{BER}(\text{SINR}_{i,j}^H)\right)^{8 \cdot B} \quad (5)$$

where $\text{BER}(\text{SINR}_{i,j}^H)$ is the bit error rate for $\text{SINR}_{i,j}^H$. If we assume that the effect of the interfering signal on the desired signal is similar to Additive White Gaussian Noise, the BER and $\text{SINR}_{i,j}^H$ can be expressed as follows [23]:

$$\text{BER}(\text{SINR}_{i,j}^H) = \frac{8}{15} \times \frac{1}{16} \times \sum_{k=2}^{16} (-1)^k \binom{16}{k} e^{\left(20 \times \text{SINR}_{i,j}^H \times \left(\frac{1}{k} - 1\right)\right)} \quad (6)$$

$$\text{SINR}_{i,j}^H = 10 \log \frac{P_{rec}(i,j)}{\sum_{h \neq j} P_{rec}(i,h) + N} \quad (7)$$

$P_{rec}(i,j)$ represents the received signal power level at device i from device j , N the noise power level, and $P_{rec}(i,h)$ the received signal power level at device i from the interfering device h that simultaneously transmits its *Keep-alive* in the *Discovery Link* under evaluation. The probability that a device i receives at least one *Keep-alive* from device j in S *Discovery Links* can be expressed as:

⁴ The mathematical model and the results obtained for $DL=1$ are equivalent to the case where $DL>1$ and $T_N = T_N/DL$ if the *Discovery Links* are uniformly distributed.

$$P_{i,j}^{H,S}|_{WH} = 1 - \prod_{s=1}^S (1 - P_{i,j}^H|_{WH}) = 1 - \prod_{s=1}^S (1 - P_j \cdot R_i \cdot PDR(SINR_{i,j}^H)) \quad (8)$$

The probability that devices i or j discover each other at least once in S *Discovery Links* can be expressed as follows:

$$P_{i \leftrightarrow j}^S|_{WH} = 1 - \prod_{s=1}^S (1 - (P_{i,j}^H|_{WH} + P_{j,i}^H|_{WH})) \quad (9)$$

where $P_{i,j}^H|_{WH}$ and $P_{j,i}^H|_{WH}$ are mutually exclusive. Finally, the discovery probability can be expressed for WH as:

$$P_{disc}|_{WH} = P_{i,j}^S \cdot P_{i \leftrightarrow j}^S|_{WH} + P_{i,j}^{S+1} \cdot P_{i \leftrightarrow j}^{S+1}|_{WH} \quad (10)$$

where $P_{i,j}^S$ and $P_{i,j}^{S+1}$ represent the probability that devices i and j are under each other's coverage during S and $S+1$ *Discovery Links*, respectively. To avoid low quality links, we define PDR_{min} as the minimum PDR needed to consider that two devices are within their coverage range and therefore are neighbors. t_{cov} represents the time during which the PDR between devices i and j is higher than PDR_{min} . In this context, S , $P_{i,j}^S$ and $P_{i,j}^{S+1}$ can be expressed as eq. (11), eq. (12) and eq. (13), respectively.

$$S = \left\lfloor \frac{t_{cov}}{T_N} \right\rfloor \quad (11)$$

$$P_{i,j}^S = \left\lfloor \frac{t_{cov}}{T_N} \right\rfloor - \frac{t_{cov}}{T_N} \quad (12)$$

$$P_{i,j}^{S+1} = \frac{t_{cov}}{T_N} - \left\lfloor \frac{t_{cov}}{T_N} \right\rfloor \quad (13)$$

2) LAN Neighbor Discovery

With LAN, the probability that a device i detects the presence of device j is equivalent to the probability that device i receives an *Advertise* transmitted by j . Since all devices are scheduled to be in reception mode when device j transmits its *Advertise*, this probability only depends on the experienced SINR:

$$P_{i,j}^H|_{LAN} = PDR(SINR_{i,j}^H) \quad (14)$$

where $SINR_{i,j}^H$ only depends on $P_{rec}(i, j)$ and N (noise power level) since there are no interfering devices on dedicated links, i.e. $P_{rec}(i, h)$ is null in eq. (7). The probability in LAN that a device i receives at least one *Advertise* in S *Advertise Links* from j is:

$$P_{i,j}^{H,S}|_{LAN} = 1 - \prod_{s=1}^S (1 - P_{i,j}^H|_{LAN}) = 1 - \prod_{s=1}^S (1 - PDR(SINR_{i,j}^H)) \quad (15)$$

The probability that devices i or j discover each other at least once in S *Management Superframes* can be expressed as:

$$P_{i \leftrightarrow j}^S|_{LAN} = 1 - \prod_{s=1}^S (1 - (1 - (1 - P_{i,j}^H|_{LAN}) \cdot (1 - P_{j,i}^H|_{LAN}))) \quad (16)$$

where $1 - (1 - P_{i,j}^H|_{LAN}) \cdot (1 - P_{j,i}^H|_{LAN})$ corresponds to the probability that devices i or j discover each other in one *Management Superframe* when one *Advertise Link* per device and *Management Superframe* is considered. $P_{i,j}^{H,S}$ and $P_{j,i}^{H,S}$ are not mutually exclusive in the case of LAN. Finally, the discovery probability can be expressed for LAN as:

$$P_{disc}|_{LAN} = 1 - (1 - (P_{i,j}^S \cdot P_{i,j}^{H,S}|_{LAN} + P_{i,j}^{S+1} \cdot P_{i,j}^{H,S+1}|_{LAN})) \cdot (1 - (P_{j,i}^S \cdot P_{j,i}^{H,S}|_{LAN} + P_{j,i}^{S+1} \cdot P_{j,i}^{H,S+1}|_{LAN})) \quad (17)$$

where $P_{i,j}^S$ and $P_{i,j}^{S+1}$ represent the probability that devices i and j are under each other's coverage range during S and $S+1$ *Advertise Links* respectively.

3) LCA Neighbor Discovery

With LCA, the probability that a given device i detects the presence of device j depends on the number of hops between i and j (Q_{ij}). This is the case because both devices will only be in reception mode during their respective *Advertise Link* if they are 1 or 2 hops away. $P_{i,j}^H$ can then be expressed for LCA as:

$$P_{i,j}^H|_{LCA} = \begin{cases} PDR(SINR_{i,j}^H) & \text{if } Q_{i,j} \leq 2 \\ 0 & \text{if } Q_{i,j} > 2 \end{cases} \quad (18)$$

The probability that device i receives at least one *Advertise* in S *Advertise Links* from j ($P_{i,j}^{H,S}$) with LCA can be obtained by combining equations (15) and (18). The probability that devices i or j detect each other at least once in S *Management Superframes* ($P_{i \leftrightarrow j}^S$) for LCA can be obtained with equations (16) and (18). The discovery probability can be expressed for LCA as:

$$P_{disc}|_{LCA} = \begin{cases} P_{disc}|_{LAN} & \text{if } Q_{i,j} \leq 2 \\ 0 & \text{if } Q_{i,j} > 2 \end{cases} \quad (19)$$

4) LANm Neighbor Discovery

LANm differentiates between static or mobile devices. As a result, the probability that a device i is able to detect a device j also depends on whether devices are mobile or static. This probability in a *Management Superframe* is:

$$P_{i,j}^H|_{LANm} = \begin{cases} P_{i,j}^H|_{WH} & \text{if } i \text{ is static} \\ 1 - (1 - P_{i,j}^H|_{WH}) \cdot (1 - P_{i,j}^H|_{LAN}) & \text{if } i \text{ is mobile} \end{cases} \quad (20)$$

The probability that a device i receives at least one *Advertise* or one *Keep-alive* in S *Management Superframes* can be obtained as:

$$P_{i,j}^{H,S}|_{LANm} = 1 - \prod_{s=1}^S (1 - P_{i,j}^H|_{LANm}) \quad (21)$$

Then, $P_{i \leftrightarrow j}^S$ can also be obtained for LANm as:

$$P_{i \leftrightarrow j}^S|_{LANm} = 1 - \prod_{s=1}^S (1 - (1 - (1 - P_{i,j}^H|_{LANm}) \cdot (1 - P_{j,i}^H|_{LANm}))) \quad (22)$$

Finally, the discovery probability for LANm also needs to take into account whether the devices are mobile or static, and can be expressed as:

$$P_{disc}|_{LANm} = \begin{cases} P_{disc}|_{WH} & \text{if } i \text{ and } j \text{ are static} \\ 1 - (1 - P_{disc}|_{WH}) \cdot (1 - P_{disc}|_{LAN}) & \text{if } i \text{ and } j \text{ are mobile} \\ 1 - (1 - P_{disc}|_{WH}) \cdot (1 - (L_S \cdot P_{i,j}^{H,S}|_{LAN} + L_{S+1} \cdot P_{i,j}^{H,S+1}|_{LAN})) & \text{if other} \end{cases} \quad (23)$$

C. Average time needed to detect a neighbor device

We consider that two devices enter into each other's communications range when the PDR is higher than PDR_{min} . In this case, t_{det} can be estimated as:

$$t_{det}|_{PROT} = \left(\frac{1}{2} + \sum_{S=1}^{\infty} ((P_{i,j}^{H,S}|_{PROT} - P_{i,j}^{H,S-1}|_{PROT}) \cdot (S-1)) \right) \cdot T_N \quad (24)$$

where PROT is the protocol considered (WH, LAN, LCA or LANm for static devices). The $\frac{1}{2}T_N$ factor corresponds to the average time elapsed from the moment two devices enter into each other's range until the first *Discovery Link* (WH and LANm static devices) or *Advertise Link* (LAN and LCA). The summation in eq. (24) corresponds to the probability of correctly receiving the first *Keep-alive* or *Advertise* in the S^{th} *Discovery Link* or *Advertise Link* from device j (i.e. $P_{i,j}^{H,S} - P_{i,j}^{H,S-1}$) multiplied by the average time elapsed until this happens (i.e. $T_N \cdot (S-1)$).

A mobile device implementing LANm can detect other devices through an *Advertise Link* or a *Discovery Link*. t_{det} is estimated for mobile devices as:

$$t_{det}|_{LANm} = \frac{1}{2} \left(\frac{T_N}{2} + \sum_{S=1}^{\infty} ((1 - P_{i,j}^{H,S}|_{LANm}) \cdot P_{i,j}^H|_{WH} \cdot (S-1) \cdot T_N) + \sum_{S=1}^{\infty} \left((1 - P_{i,j}^{H,S}|_{LANm}) \cdot (1 - P_{i,j}^H|_{WH}) \cdot P_{i,j}^H|_{LAN} \cdot \left(S - \frac{1}{2} \right) \cdot T_N \right) \right) \\ + \frac{1}{2} \left(\frac{T_N}{2} + \sum_{S=1}^{\infty} ((1 - P_{i,j}^{H,S}|_{LANm}) \cdot P_{i,j}^H|_{LAN} \cdot (S-1) \cdot T_N) + \sum_{S=1}^{\infty} \left((1 - P_{i,j}^{H,S}|_{LANm}) \cdot (1 - P_{i,j}^H|_{LAN}) \cdot P_{i,j}^H|_{WH} \cdot \left(S - \frac{1}{2} \right) \cdot T_N \right) \right) \quad (25)$$

The first line (i.e. the first term of the sum in eq. (25) that is multiplied by the factor $\frac{1}{2}$) corresponds to the case in which the mobile device tries first to detect a device in a *Discovery Link*. The time to detect a device is then equal to the sum of the probability of correctly receiving the S^{th} *Keep-alive*, multiplied by the average time elapsed until then $((S-1) \cdot T_N)$, and of the probability of not receiving the S^{th} *Keep-alive* multiplied by the probability of correctly receiving the S^{th} *Advertise* and by the average time elapsed until then $((S-\frac{1}{2}) \cdot T_N)$. The second line (i.e. the second term of the sum in eq. (25) that is multiplied by the factor $\frac{1}{2}$) corresponds to the case in which the mobile device tries first to detect a device in one *Advertise Link*. The expressions are obtained similarly to the case in which the mobile device tries first to detect neighbor devices in a *Discovery Link*, but replacing *Keep-alive* by *Advertise*.

D. Efficiency

In WH, each *Discovery Link* has an associated *NBC* of $\rho \cdot C$ bytes, where C is the number of frequency channels and ρ represents the amount of information that can be transmitted in a link. In this study, we consider ρ equal to 133 bytes, since 133 is the maximum number of bytes that can be transmitted in a link. The *NBC* associated to WH (eq. (26)) includes ρ bytes resulting from the transmission of a *Keep-alive* in a *Discovery Link*, and $\rho \cdot (C-1)$ bytes since all other devices need to be in reception mode during the *Discovery Link* and therefore cannot utilize the remaining $C-1$ channels.

$$NBC|_{WH} = \rho + \rho \cdot (C - 1) = \rho \cdot C \quad (26)$$

LAN utilizes H *Advertise Links* when the network has H devices. Each *Advertise Link* requires an *NBC* of $\rho \cdot (C-1)$ bytes because all devices need to be in reception mode during this link, and therefore the remaining $C-1$ channels cannot be used. The *NBC* of LAN does not take into account the transmission of an *Advertise* because it is needed to provide network information to devices trying to join the network, and therefore it has to be transmitted independently of the NDP. On the other hand, we need to take into account the *NBC* consumed by the *Network Manager* to schedule each device to be in reception mode in $H-1$ *Advertise Links*. In this context, the average *NBC* that LAN generates per *Management Superframe* is then:

$$NBC|_{LAN} = H \cdot \rho \cdot (C - 1) + \frac{T_N}{t} \cdot \sum_{h=1}^H (H - 1) \cdot B_D \cdot hop_h \quad (27)$$

where t is the elapsed time, T_N is the duration of a *Management Superframe*, B_D represents the number of bytes transmitted by the *Network Manager* to send the schedule of an *Advertise Link* to one device, and hop_h is the number of hops between device h and the *Network Manager*.

With LCA, only two-hop neighbor devices need to be in reception mode for each device's *Advertise Link*. The rest of devices can utilize the other channels that are not used to transmit the given *Advertise Link*. Similarly to LAN, the *Network Manager* also has to schedule all the *Advertise Links* with LCA. Additionally, each device needs to include in its *Advertise* a list with the identity of its one-hop neighbors. The *NBC* associated to LCA can be computed as:

$$NBC|_{LCA} = \sum_{h=1}^H H_{h1} \cdot B_A + \frac{T_N}{t} \cdot \sum_{h=1}^H (H - 1) \cdot B_D \cdot hop_h + \rho \cdot \sum_{\forall h \in H^*} ((C - 1) - (H - H_{h2} - 1)) \quad (28)$$

where H_{h1} is the number of one-hop neighbors of device h , B_A is the amount of bytes sent in each *Advertise* per device, and H_{h2} is the number of one- or two-hop neighbors of device h . The first term of the equation represents the additional information sent by all devices in their respective *Advertise* to include the identity of their one-hop neighbors. This information is transmitted by all devices in every *Management Superframe*. The second term in eq. (28) refers to the *NBC* associated to the scheduling process conducted by the *Network Manager*, which is exactly the same as in LAN. The third term in eq. (28) quantifies the number of bytes consumed considering that only one-hop and two-hop neighbor devices need to be in reception mode for each device's *Advertise Link*. Devices at more than two hops from a device sending an *Advertise* could be in transmission or reception mode in other channels for functions not related to neighbor discovery. The number of channels that cannot be used when a given device h transmits its *Advertise* is $(C-1) - (H - H_{h2} - 1)$, where $H - H_{h2} - 1$ is the number of devices at more than two hops from device h and $C-1$ is the number of available channels for other functions. The *NBC* associated to the transmission of the *Advertise* by device h is then $\rho \cdot (C-1) - (H - H_{h2} - 1)$ if $C-1$ is higher than $H - H_{h2} - 1$. If $C-1$ is lower or equal than $H - H_{h2} - 1$, then all available channels can be utilized for functions not related to neighbor discovery, and the *NBC* associated to this process is null. To include this effect in equation (28), only nodes $h \in H^*$ are considered in the summation, where H^* is the set of devices for which $(C-1) > (H - H_{h2} - 1)$.

All devices implementing LANm utilize the transmission of *Keep-alive* in the *Discovery Link* for the discovery process. This has an associated *NBC* of $\rho \cdot C$ bytes. Additionally, each of the M mobile devices is scheduled by the *Network Manager* to be in reception mode in $H-1$ *Advertise Links* so that they can receive the *Advertise* from the other network devices. The average *NBC* associated to LANm per *Management Superframe* can then be estimated as:

$$NBC|_{LANm} = \rho \cdot C + \frac{T_N}{t} \cdot \sum_{m=1}^M (H-1) \cdot B_D \cdot hop_m \quad (29)$$

E. Energy consumption

Eq. (30) represents the average energy consumed per device and per *Management Superframe* by WH. It is computed considering the transmission and reception of *Keep-alive* in each *Discovery Link*. P_i (eq. (2)) and R_i (eq. (3)) represent the probability of transmitting and receiving a *Keep-alive* in each *Discovery Link*, respectively. E_{Tx-KA} and E_{Rx-KA} represent the energy consumed by a device when transmitting and receiving a *Keep-alive*, respectively. P_{i-NRx} represents the probability that the received signal strength of a *Keep-alive* is below the minimum signal needed to start the reception process. In this case, the device goes to idle mode and consumes E_{idle} ⁵.

$$E_d|_{WH} = \frac{T_N}{H \cdot t} \cdot E_C + \frac{1}{H} \cdot \sum_{i=1}^H (P_i \cdot E_{Tx-KA} + (R_i - P_{i-NRx}) \cdot E_{Rx-KA} + P_{i-NRx} \cdot E_{idle}) \quad (30)$$

The average energy consumed by LAN (eq. (31)) does not consider the energy consumed by devices transmitting *Advertise* (E_{Tx-Ad}) since they have to be transmitted anyway to help new devices join the network. The energy is computed considering the reception of *Advertise* (E_{Rx-Ad}) and the energy consumed when a device goes into idle mode. $P_{i-Rx-Ad(j)}$ represents the probability that device i receives an *Advertise* transmitted by device j with a signal strength above the minimum threshold level needed to start the reception.

$$E_d|_{LAN} = \frac{T_N}{H \cdot t} \cdot E_C + \frac{1}{H} \cdot \sum_{i=1}^H \sum_{j \neq i}^H (P_{i-Rx-Ad(j)} \cdot E_{Rx-Ad} + (1 - P_{i-Rx-Ad(j)}) \cdot E_{idle}) \quad (31)$$

The average energy consumed by LCA is expressed in eq. (32). It is computed considering the energy consumed by the transmission of the list of one-hop neighbors attached to each *Advertise* (E_{list}), and the energy that a device i consumes when listening or receiving the *Advertise* of its two-hop neighbors. The number of one- and two-hop neighbors is represented by H_{i2} .

$$E_d|_{LCA} = \frac{T_N}{H \cdot t} \cdot E_C + \frac{1}{H} \cdot \sum_{i=1}^H \left(E_{list} + \sum_{j \neq i}^{H_{i2}} (P_{i-Rx-Ad(j)} \cdot E_{Rx-Ad(j)} + (1 - P_{i-Rx-Ad(j)}) \cdot E_{idle}) \right) \quad (32)$$

The average energy consumed by LANm (eq. (33)) is equal to the energy consumed by WH (all devices use *Keep-alive* for the discovery process) plus the energy consumed by M mobile devices when listening *Advertise Links*.

$$E_d|_{LANm} = E_d|_{WH} + \frac{T_N}{H \cdot t} \cdot E_C + \frac{1}{H} \cdot \sum_{i=1}^M \sum_{j \neq i}^H (P_{i-Rx-Ad(j)} \cdot E_{Rx-Ad} + (1 - P_{i-Rx-Ad(j)}) \cdot E_{idle}) \quad (33)$$

7. Performance Evaluation

A. Simulation platform

The performance of the different neighbor discovery protocols is evaluated using a simulation platform implemented in Matlab. The platform implements in detail all the functions that intervene in the neighbor discovery process, including the necessary functions at the *Network Manager*, a detailed radio propagation model, a mobility model, and the main modules of the physical and MAC (Medium Access Control) layer of the WirelessHART protocol stack. All the modules have been implemented following the WirelessHART standard and state-of-the-art simulation practices (e.g. for modeling the propagation and the interference) and have been validated in detail. Fig. 1 shows the main components of the implemented simulation platform. A description of the main modules is presented below.

⁵ The energy consumption is estimated based on the WirelessHART standard, which establishes that when the received signal strength of a DLPDU is below the minimum signal needed to start the reception process, the receiver turns off the radio after 2ms (idle mode).

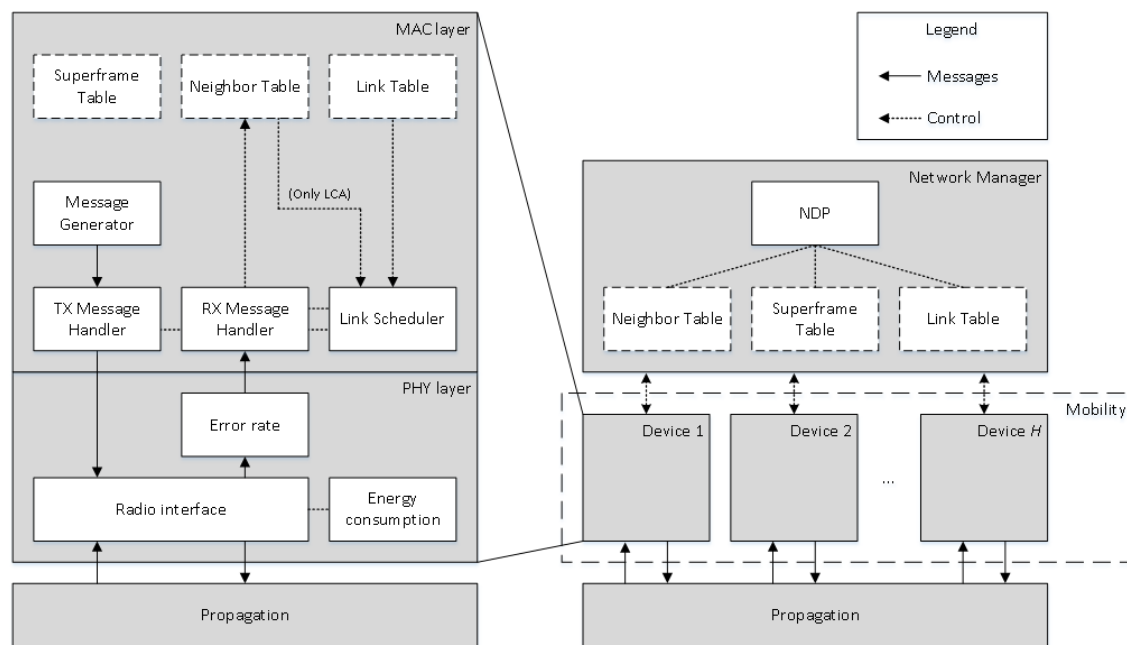


Fig. 1. Block diagram of the simulation platform implemented

To account for realistic propagation conditions, the platform implements the radio propagation model proposed in [25] for industrial environments. The model considers a one-slope log-distance path-loss model, and takes into account the fading effects at 2.4GHz using a log-normal random distribution. Some of the main parameters of the model are: $d_0=15\text{m}$ (reference distance), $PL(d_0)=71.84\text{dBm}$ (pathloss at d_0 distance), $n=2.16$ (pathloss exponent), $\sigma=8.13\text{dBm}$ (shadowing standard deviation), $G_t=G_r=0\text{dB}$ (transmitter and receiver antenna gains), and $N=-93\text{dBm}$ (average noise power).

At the physical layer, the simulation platform operates at the packet level and probabilistically determines if a packet is correctly received or not based on its SINR (Signal to Interference and Noise Ratio) and an error rate model. This model emulates the performance of the physical layer of IEEE 802.15.4, which is the physical layer adopted in WirelessHART. To determine whether a packet is correctly received or not, a receiver's radio interface computes the SINR. In dedicated time slots, only one packet can be transmitted in the network, and it does not suffer any interference. In this case, the radio interface computes the SINR of the received packet using the received power level (estimated with the propagation model) and the average noise power level. In shared time slots, multiple packets can simultaneously arrive to the receiver and produce interference. In this case, the radio interface identifies the packet with highest received power level as the candidate packet to be received, and the rest of packets are considered to represent interference and are used to calculate the SINR. The simulation platform implements the error rate model proposed in the IEEE 802.15.4 standard for AWGN channels at 2.4GHz [23]. In particular, the model utilized is the one obtained for the OQPSK modulation that is used in WirelessHART [3].

The physical layer of the implemented simulation platform is also in charge of computing the energy consumed for the transmission and reception of packets. To do so, the platform models the time slot structure specified in the WirelessHART standard. The simulation platform models the power consumption using the reference values specified for the XDM2510H transceiver when it is powered with 3.6V [26]. This results in a power consumption of 18mA when the transceiver is in transmission mode and 6mA when the transceiver is in reception mode. The transmission power of the XDM2510H transceiver is 8dBm, and it has a reception sensibility of -90dBm.

The implementation of the MAC layer is also in line with the WirelessHART standard, and includes all the modules that influence the neighbor discovery process. The MAC layer implemented in the simulation platform includes the following three tables that control the communications of each device and collect communication statistics as specified in the WirelessHART standard:

- *Superframe Table*: it stores the attributes of a superframe, for example, the number of slots and the unique identifier.
- *Link Table*: it stores the attributes of all links including the type of link (normal, broadcast, join or discovery), the slot number in the superframe assigned to the link, the link use (transmission, reception or shared), the channel offset, and the neighbor ID (for dedicated links) among others.
- *Neighbor Table*: it stores the ID of all devices with which the device may be able to directly communicate with (1-hop neighbors). In LCA, this table has been extended to also include 2-hop neighbors.

The implementation of the *Link Scheduler* also follows the WirelessHART standard. Each device has a link scheduler that consults the tables of the device in order to identify the next slot that should be used for listening or sending a message. As it can be observed

in Fig. 1, the *Link Scheduler* is connected to the Tx and Rx Message Handler modules that are in charge of transmitting and receiving messages to/from the physical layer. For a given device, only one of them (Tx or Rx) can be active at the same time. The *Tx Message Handler* passes to the physical layer the messages generated at the MAC by the *Message Generator* together with the information about the channel frequency to be used. The *Rx Message Handler* informs the physical layer of the channel frequency utilized, receives the messages that are successfully decoded by the physical layer, and updates the neighbor table based on the received messages. Differences can be found at the *Rx Message Handler* depending on the neighbor discovery protocol:

- WH only updates the *Neighbor Table* upon the reception of *Keep-alive* messages. In fact, devices that belong to the network are not scheduled by the network manager to listen to *Advertise* messages.
- LAN and LCA update the *Neighbor Table* upon the reception of *Advertise* messages. In fact, devices do not need to be configured to listen to *Keep-alive* messages.
- LANm updates the *Neighbor Table* upon the reception of *Advertise* and *Keep-alive* messages. Mobile devices are scheduled to transmit and receive *Advertise* messages, but static devices run the WirelessHART neighbor discovery protocol (i.e. they exploit *Keep-alive* messages).

The operation of the link scheduler mainly depends on the information in the tables that are configured by the *Network Manager*. As a result, the operation of the implemented *Link Scheduler* depends on the neighbor discovery protocol:

- WH: the *Link Scheduler* is in charge of managing the timer used to randomly transmit or receive a *Keep-alive* in *Discovery Links*. To this aim, the *Link Scheduler* gets from the *Link Table* the information about the existing *Discovery Links*.
- LAN: the *Link Scheduler* gets from the *Link Table* the information about the links where *Advertise* need to be transmitted and received.
- LANm: the *Link Scheduler* of mobile devices also follows the information in the *Link Table* for the transmission and reception of *Advertise*. The *Link Scheduler* of static devices operates following WH.
- LCA: the *Link Scheduler* gets from the *Link Table* the information about the *Advertise* transmitted by all devices in the network. However, it only listens to those devices that are at one or two hops. To this aim, LCA needs to have access to the information on the *Neighbor Table* (Fig. 1). The ID of the one-hop neighbors is attached to the corresponding *Advertise* by the *Tx Message Handler*.

The WirelessHART *Network Manager* is in charge of allocating all links in the *Management Superframe*. To this aim, the *Network Manager* configures the *Link Table* of all devices. In the implemented simulation platform, the *Network Manager* allocates the links for the neighbor discovery process following the protocol under evaluation:

- WH: the *Network Manager* allocates the *Discovery Links* for the transmission of *Keep-alive* messages. *Discovery Links* are shared by all devices and therefore collisions can take place. Once these links have been allocated, the *Link Scheduler* of each device randomly selects to transmit a *Keep-alive* or stay in reception mode in each *Discovery Link*.
- LAN: the *Network Manager* randomly allocates the *Advertise Links* of all devices, and avoids that two devices transmit their *Advertise* in the same slot. *Advertise Links* are dedicated links. As a result, each device has its own *Advertise Link* to transmit its *Advertise* message. The *Network Manager* configures the *Link Table* of all devices so that each one of them can listen to the *Advertise* sent by the other devices.
- LCA: the *Network Manager* executes the same allocation as when implementing LAN. However, the final decision to listen or not a given *Advertise* is taken locally by each device based on its *Neighbor Table*. A device will be in reception mode only for the *Advertise* of its one-hop and two-hop neighbors.
- LANm: the *Network Manager* allocates the *Discovery Links* for the transmission of *Keep-alive* messages following the WH protocol. In addition, it allocates the transmission of *Advertise Links* in dedicated links for all devices. Only mobile devices are configured to be in reception mode when all devices (mobile and static) send their *Advertise*.

Given the objectives of this study, the simulation platform does not implement the mechanisms needed by the *Network Manager* to send to each device its *Link Table*. We assume that these tables are configured and sent to each device when they join the network. They do not need to be updated, and as a result they do not influence this study that is focused on the performance and efficiency of neighbor discovery protocols.

By default, the WH and LANm NDPs have been evaluated considering one *Discovery Link* in each *Management Superframe* and $DLN_i=10$. This value was selected based on preliminary simulations that showed that it generally maximizes $P_{i,j}^H$ for topologies with more than 16 devices. For LCA, B_A has been set equal to 2 bytes (the ID of each device), and B_D has been set equal to 5 bytes (2 bytes of the device ID, 1 byte for the WirelessHART channel list offset, and 2 bytes for the slot offset in the *Management Superframe*). A sufficiently high number of simulations have been conducted for each NDP and scenario under evaluation in order to ensure the statistical accuracy of the results, and in particular a relative error below 1% for all results reported in this paper.

B. Evaluation Scenarios

This work considers an industrial wireless sensor network scenario with a grid topology of static networked sensors/actuators. The grid of static devices supports the connection of mobile devices to the network [27]. Three grid topologies with different number of static devices (16, 25 and 36) have been analyzed (see Fig. 2). For each topology, we analyze two different densities of static devices by varying the d parameter in Fig. 2 ($d=30\text{m}$ and $d=60\text{m}$). The *Network Manager* (depicted in black in Fig. 2) is located in the position that minimizes the average number of hops to all static devices.

Mobile devices have been deployed following two use cases. The first one considers that up to 5 mobile devices move following a random waypoint trajectory around the grid of static devices (see Fig. 3a). To this aim, each mobile device selects a random destination point and a random speed between 0.1 and 3m/s. When the mobile device reaches the selected destination point, it repeats the process and starts moving towards a new random destination point without any pause. The second use case considers a single mobile device that moves along a corridor in the grid of static devices (see Fig. 3b). The mobile device could represent a person, a vehicle or a robot that needs to periodically transmit data packets towards certain destination device. This use case has been considered to evaluate the probability that a mobile device remains connected to the network (P_{conn}). In this use case, the mobile device moves from *Start* to *End* following the red arrow shown in Fig. 3b. The speed (v) of the mobile device is randomly chosen between 0.1m/s and 3m/s.

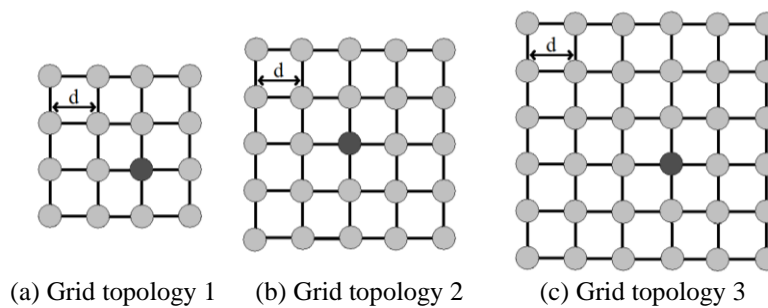


Fig. 2. Evaluation topologies

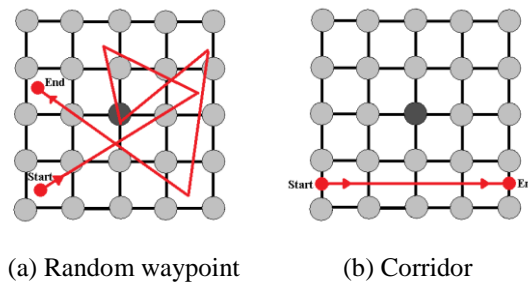


Fig. 3. Mobility use cases over the grid topology 2

C. Validation of Analytical Models

The performance of the proposed NDPs has been evaluated through extensive simulations following the scenarios and settings described in the previous section. This has been the case to account for realistic mobility, deployment and operating conditions that are difficult to integrate into the analytical models. The analytical performance models developed in Section 6 represent though valuable tools to obtain first insights into the performance of NDPs. This is demonstrated in this section where the results obtained with the models are compared with simulation results. This comparison is done under grid topology 2, and considering mobile devices moving from *Start* to *End* position following the red arrow in Fig. 2. We assume for this analysis that t_{cov} is the time a mobile device needs to go from *Start* to *End*. We consider for the analytical performance models that a DLPDU is not correctly received if two or more devices simultaneously transmit on the same link (independently of the interference level). Fig. 4 represents the discovery probability (P_{disc}) between the mobile device and *Network Manager*. The results obtained for WH, LAN and LANm⁶ show a close match between the results obtained using the analytical performance models (*Ana*) and the simulations (*Sim*). These results validate the proposed models and their usability to obtain first insights into the performance trends of NDPs.

⁶ LCA is only analyzed through simulations due to its discovery process.

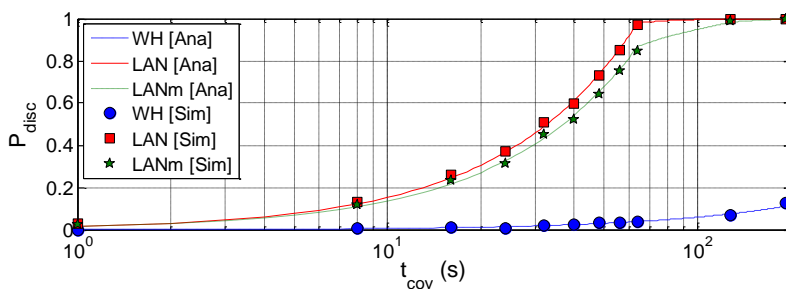


Fig. 4. Discovery probability (P_{disc}) for grid topology 2 with $d=30\text{m}$ and $T_N=64\text{s}$.

D. Simulation Results

Fig. 5 represents the probability that a given device i discovers another device j or vice versa (P_{disc}) for a range of t_{cov} values. These results are obtained in grid topology 1 with mobile devices following a random waypoint trajectory and considering $T_N=64\text{s}$ (left) and $T_N=16\text{s}$ (right). $T_N=64\text{s}$ is the default duration of the WirelessHART *Management Superframe*. The obtained results show that the protocols that exploit the transmission of *Advertise* for the discovery process achieve similar results. Their performance is significantly higher than that achieved with the standard WH protocol. The performance of the LANm protocol is slightly lower than that of LAN and LCA. This is due to the fact that static devices implementing LANm only make use of *Discovery Links* to discover neighbor devices, but LAN and LCA utilize *Advertise Links* for both static and mobile devices. It is interesting to note that LAN and LCA present nearly the same performance despite the fact that LCA devices are only scheduled to listen for their two-hop neighbor *Advertise*. Similar trends to those reported in Fig. 5 have been observed for different PDR_{min} values, grid topologies and node densities.

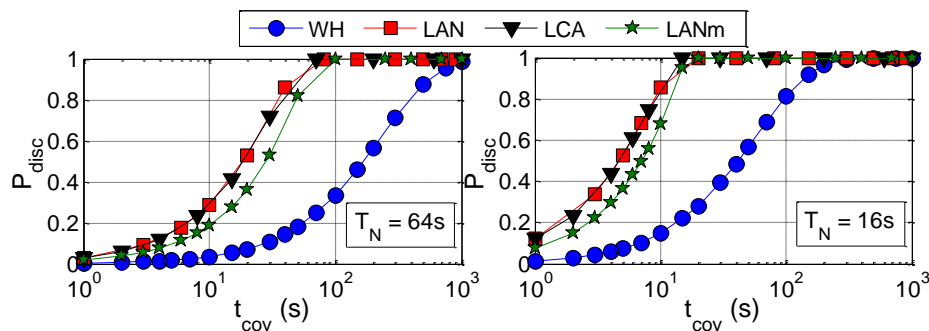


Fig. 5. Discovery probability (P_{disc}) with $PDR_{min}=0.95$, $d=60\text{m}$ and $M=1$ in grid topology 1.

Table 2 presents the time required to detect a neighbor (t_{det}) as a function of T_N considering a single mobile device ($M=1$) in grid topologies 1, 2 and 3 under realistic propagation and interference conditions, and considering again mobile devices with random trajectories. The table shows average values and the 99th percentiles in order to identify upper performance bounds considering propagation and interference effects and the fact that WH is based on a probabilistic discovery scheme. Table 2 shows that the average time needed by WH to detect a neighbor is around 25 times higher than the time needed by LAN, LCA and mobile devices that employ LANm⁷. The simulated scenarios consider a significant larger portion of static devices. This explains why when averaging the time needed to detect neighbors for all LANm devices the performance levels are close to those observed with WH. Table 2 also shows that the time increases for WH with the number of total devices in the scenario due to the interference levels experienced in the *Discovery Links*. The increase is not significant for LAN, LCA and mobile devices employing LANm. These trends are consistent with those that can be inferred from eq. (24) since protocols with lower $P_{i,j}^{H,S}$ (i.e. WH) need more time to detect a neighbor. Table 2 also depicts the results that could be obtained with WH when considering a *Discovery Link* per static or mobile device (referred to as WH^{*}). This allows comparing a probabilistic NDP scheme (WH) with deterministic NDP schemes (LAN, LCA y LANm) using the same number of radio resources or links for the discovery process. In this case, WH^{*} can significantly reduce the average time to detect a neighbor. However, this is obtained at the cost of a significant increase in the number of radio resources exclusively reserved for the discovery process (e.g. WH^{*} requires 37 *Discovery Links* in grid topology 3). This is in contrast to the deterministic schemes that achieve similar performance levels to those observed with WH^{*}, but using

⁷ As an example, the average time employed by WH to detect a neighbor device is around 12 minutes considering a superframe of 64s and a mobile device speed of 1m/s in grid topology 1. In 12 minutes, the distance travelled by the mobile device is 720m. The neighbors of the mobile device at position 0m and at position 720m will totally differ. With the proposed protocols (LAN, LCA and LANm), the distance travelled would be reduced to around 35m.

a lower number of radio resources.

Table 2 - Time to detect a neighbor t_{det} (s)

Metric	Grid topology	WH	WH*	LAN	LCA	LANm (All devices)	LANm (only mobile devices)
Average	1	$11.378 \cdot T_N$	$0.669 \cdot T_N$	$0.518 \cdot T_N$	$0.523 \cdot T_N$	$10.738 \cdot T_N$	$0.495 \cdot T_N$
	2	$12.811 \cdot T_N$	$0.493 \cdot T_N$	$0.518 \cdot T_N$	$0.524 \cdot T_N$	$12.337 \cdot T_N$	$0.497 \cdot T_N$
	3	$14.274 \cdot T_N$	$0.386 \cdot T_N$	$0.518 \cdot T_N$	$0.527 \cdot T_N$	$13.902 \cdot T_N$	$0.499 \cdot T_N$
99th percentile	1	$54.284 \cdot T_N$	$3.193 \cdot T_N$	$1.438 \cdot T_N$	$1.561 \cdot T_N$	$53.692 \cdot T_N$	$1.365 \cdot T_N$
	2	$62.959 \cdot T_N$	$2.422 \cdot T_N$	$1.438 \cdot T_N$	$1.597 \cdot T_N$	$62.708 \cdot T_N$	$1.369 \cdot T_N$
	3	$71.691 \cdot T_N$	$1.938 \cdot T_N$	$1.438 \cdot T_N$	$1.636 \cdot T_N$	$71.058 \cdot T_N$	$1.378 \cdot T_N$

Fig. 6 illustrates the probability that a mobile device remains connected to the network while it moves from *Start* to *End* position along a corridor (P_{conn}). The probability is depicted as a function of the speed of the mobile device (v). For a fair comparison, WH and LANm are here configured with the DLN_i value that maximizes P_{conn} ($DLN_i=3$ for WH and $DLN_i=6$ for LANm). This value differs from the one that maximizes $P_{i,j}^H$. The obtained results show that WH is not able to maintain the mobile node connected to the network as it moves along the corridor, even under low speed values. On the other hand, the protocols that exploit *Advertise Links* to discover neighbor devices can better maintain the network connectivity of mobile devices. This is specially the case for low T_N values as they result in a higher number of *Keep-alive* and *Advertise* sent and received per unit of time. The probability that a mobile device remains connected to the network while moving around the scenario can be improved with higher densities of static devices. Fig. 7 illustrates P_{conn} for $d=30m$ and grid topology 1. The comparison of Fig. 6 and Fig. 7 shows how P_{conn} increases for WH with higher densities. However, even with higher densities, WH can only successfully maintain the connectivity when mobile devices move at low speeds. A higher density of static devices also helps LAN, LCA and LANm to overcome the limitations resulting from using long *Management Superframes* ($T_N=64s$).

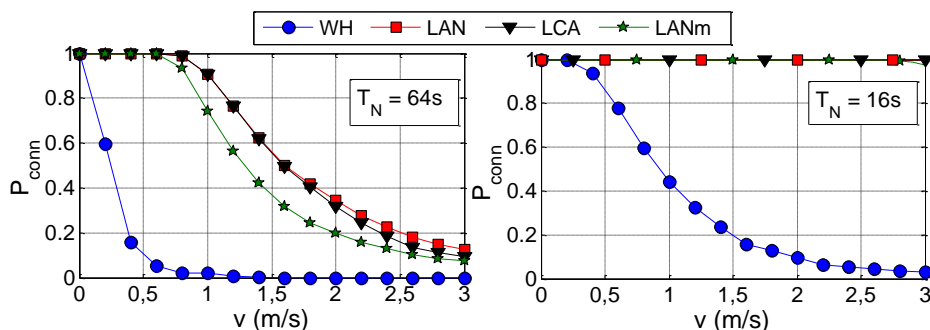


Fig. 6. P_{conn} as a function of the speed of the mobile device (v). Grid topology 1 with $d=60m$ and $PDR_{min}=0.95$.

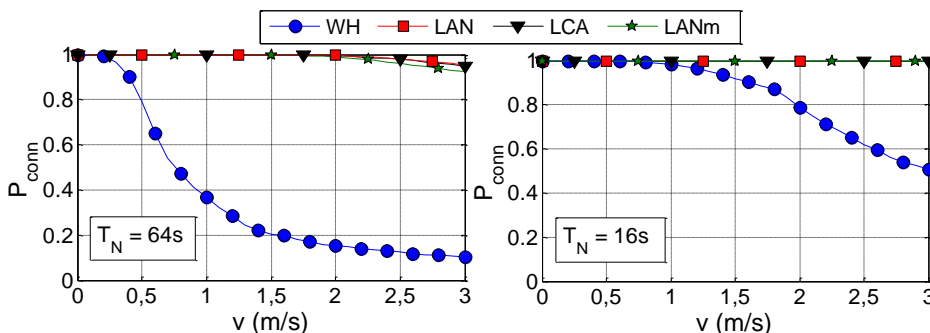


Fig. 7. P_{conn} as a function of the speed of the mobile device (v). Grid topology 1 with $d=30m$ and $PDR_{min}=0.95$.

Fig. 8 represents P_{conn} as a function of the speed of the mobile device while it moves from *Start* to *End* position along a corridor for different PDR_{min} values. All protocols improve P_{conn} when PDR_{min} is reduced since devices can discover each other at larger distances. However, reducing PDR_{min} decreases the minimum link quality required to discover a neighbor, and therefore decreases the average QoS. WH again shows the lowest performance compared to the protocols that exploit *Advertise* to discover neighbor devices. Higher probabilities can be obtained with $T_N=16s$ following the trends exhibited in Fig. 6 and Fig. 7.

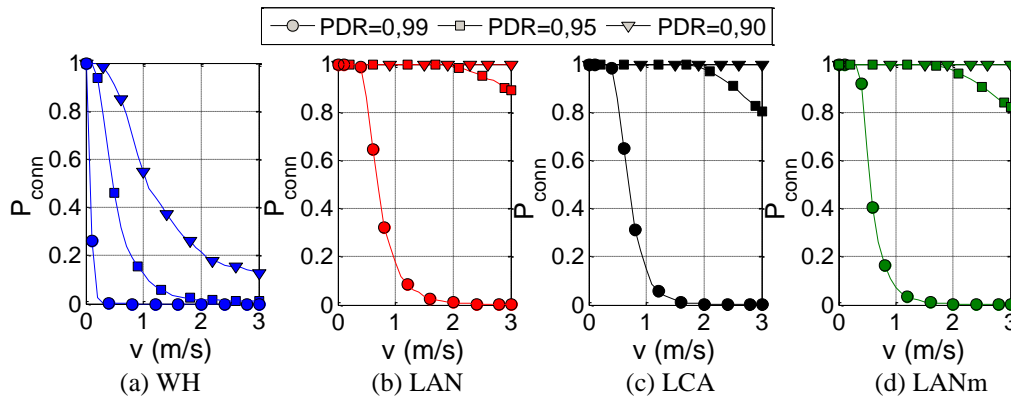


Fig. 8. P_{conn} as a function of the speed of the mobile device (v). Grid topology 3 with $T_N=64s$ and $d=30m$.

The previous results have shown that the protocols that exploit *Advertise* to detect neighbor devices generally achieve similar performance levels. The differences between the protocols appear when analyzing their efficiency. Fig. 9 represents the average *NBC* per *Management Superframe* as a function of the elapsed time t when considering mobile devices that move following a random waypoint trajectory. The results shown in Fig. 9 correspond to the average *NBC* values obtained for all simulated scenarios (including the various d , M and PDR_{min} values analyzed in the study). Part of the *NBC* generated by the proposed protocols is only produced in the initial *Management Superframe* (to schedule devices). As a result, the average *NBC* decreases with the elapsed time. Fig. 9 shows that the *NBC* for WH depends on the number of active channels but not on the elapsed time. LANm results in an *NBC* very similar to that of WH (especially for large values of t) since both protocols only differ on the operation of mobile devices, and the ratio of mobile to static devices is low. This effect can also be observed from the derived analytical performance models, in particular eq. (28) and (29). It is interesting to note that for large values of t , LCA offers the lowest total *NBC*. This is the case because for large values of t the *NBC* generated at the initial *Management Superframe* to schedule all *Advertise* becomes negligible (second term in eq. (28)), and the main contributions to the total *NBC* are the additional information (2 Bytes per neighbor) introduced in all *Advertise* (first term in eq. (28)) and the *NBC* resulting from the fact that one-hop and two-hop neighbors need to be in reception mode to receive the corresponding *Advertise* (third term in eq. (28)). This results in a lower total *NBC* for LCA than e.g. WH for large values of t . The *NBC* for WH is independent of t and takes into account that all devices need to be in reception mode during the *Discovery Link*, thus blocking the use of all channels during the *Discovery Link* for other functions. Fig. 9 shows that LAN results in the highest *NBC* since it requires all devices to listen for each other's *Advertise*, and the use of other frequency channels in the time slots where *Advertise* are transmitted is hence blocked.

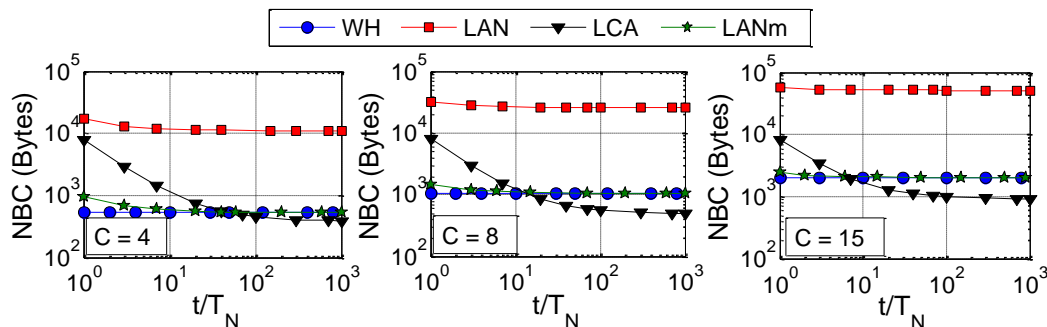


Fig. 9. Average *NBC* as a function of the elapsed time t for different number of active frequency channels C .

Table 3 presents the total number of links used for neighbor discovery per *Management Superframe* (*Total* in Table 3). This includes the *Advertise Links* for LAN and LCA, the *Advertise* and *Discovery Links* for LANm, and the *Discovery Links* for WH. Table 3 also shows the number of dedicated *Discovery Links* that nodes utilize to transmit packets used in the discovery process. Finally, Table 3 includes the number of links during which each device needs to be in reception mode to receive a packet used in the discovery process (*Reception*). This is equal to the number of *Discovery Links* for WH and the static devices in LANm. When using LAN, each device must be in reception mode for the $H-1$ *Advertise Links* of the rest of $H-1$ network devices. With LCA, each device must only be in reception mode for the *Advertise Links* of its one- or two-hop neighbors (H_{h2}). Mobile devices in LANm must be in reception mode for the *Discovery Link* and the $H-1$ *Advertise Links* of the rest of $H-1$ network devices. Table 3 shows that the proposed techniques (LCA and LANm) use more links than WH for the discovery process. However, they do not dedicate exclusively for the discovery process more radio resources than WH (LCA even less). Instead, they use other radio resources (*Advertise Links*) that already exist in WirelessHART, which improves the utilization of radio resources.

Table 3 – Links for neighbor discovery per *Management Superframe*

	WH	LAN	LCA	LANm (static)	LANm (mobile)
Total	1	H	H	1	$1+H$
Dedicated <i>Discovery Links</i>	1	0	0	1	1
Reception	1	$H-1$	H_{h2}	1	H

Table 4 compares the energy consumed per device (E_d) by the different protocols for each grid topology and different number M of mobile devices. The results have been obtained considering 1000 *Management Superframes* ($t/T_N = 1000$). The results show that the average energy consumed per device increases with the number of devices since devices need to listen and receive more *Advertise* or *Keep-alive*. The results obtained also show that WH presents the lowest average energy consumption. Only LANm has an energy consumption level similar to that achieved by WH when no mobile devices are deployed. The deployment of mobile devices increases the average energy consumption, especially for LAN and LCA. As it could be foreseen from eq. (30)-(33), the highest energy consumption is obtained with LAN since LCA and LANm reduce the number of devices that actively participate in the discovery process. LANm is the proposed protocol that achieves the lowest energy consumption. LCA and LANm significantly reduce the energy consumption with respect to LAN (by 50% and 96% respectively). WH consumes less energy for the discovery process. However, LCA and LANm significantly improve the probability to detect neighbor devices and the time to do so. It is also important noting that the energy consumed by the deterministic NDP schemes in the discovery process is relatively small. For example, the energy consumed by LAN during the discovery process in grid topology 3 is equivalent to 0.03% of the energy a node would consume if transmitting and receiving packets continuously.

Table 4 – Average energy consumed per device and per *Management Superframe* E_d (μJ)

Grid topology	M	WH	LAN	LCA	LANm
1	0	54.1	684.0	543.5	54.1
	1	54.2	728.9	570.2	97.0
	5	54.7	906.5	650.3	269.1
2	0	54.4	1077.5	700.7	54.4
	1	54.5	1121.6	716.9	97.6
	5	54.8	1297.3	770.7	270.3
3	0	54.8	1558.9	867.1	54.8
	1	54.9	1602.7	877.9	98.1
	5	55.0	1777.5	915.2	271.5

8. Conclusions

This study has proposed and evaluated novel neighbor discovery protocols for industrial wireless sensor networks that include mobile devices. The proposed protocols exploit information about the network topology and the nature of devices (static or mobile) to improve the capacity to detect neighbor devices while efficiently utilizing the radio resources (referred to as links in WirelessHART). The proposed protocols have been developed for the WirelessHART standard given its important industrial adoption. However, it should be noted that the proposed schemes can be adapted for the ISA100.11a and IEEE 802.15.4.e standards since they maintain WirelessHART's passive listening scheme for neighbor discovery, and introduce a scanning scheme with active solicitations for the advertisements.

This study has also presented accurate analytical performance models for all the neighbor discovery protocols under evaluation, including the protocol embedded in the WirelessHART standard. These models easily provide first insights into the performance and operation of the proposed protocols. A thorough evaluation of the neighbor discovery protocols has then been conducted through simulations to account for realistic mobility, radio propagation, deployment and operating conditions. The conducted evaluation has shown that the proposed NDP protocols significantly improve the capacity of mobile devices to discover neighbor devices and the time needed to detect them compared to the existing WirelessHART NDP solution. In particular, the proposed NDP protocols result in that mobile devices can detect neighbor devices up to 25 times faster than with the WirelessHART NDP protocol. These benefits are generally achieved at the expense of a lower efficiency or higher energy consumption compared to the current WirelessHART NDP process. The results obtained show that the LANm proposal achieves the best trade-off between performance, efficiency and energy consumption. This proposal significantly improves the capacity of mobile devices to detect neighbor devices, and achieves energy consumption and efficiency levels very close to those obtained with the WirelessHART NDP protocol.

Acknowledgments

This work was supported in part by the Spanish Ministry of Economy and Competitiveness and FEDER funds under the project TEC2014-57146-R, by the Local Government of Valencia with reference ACIF/2013/060 and by the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No 723909 (AUTOWARE project).

References

- [1] 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, Mar. 2014.
- [2] A. Willig, "Recent and Emerging Topics in Wireless Industrial Communications: A Selection," *IEEE Transactions on Industrial Informatics*, vol. 4, pp. 102-124, May 2008.
- [3] IEC 62591 Ed. 1.0: Industrial communication networks -Wireless communication network and communication profiles- WirelessHART™, International Electrotechnical Commission, IEC, 2010.
- [4] Wireless Systems for Industrial Automation: Process Control and Related Applications, ISA-100.11a-2009 Standard, 2009.
- [5] IEEE Std. 802.15.4e-2012, "Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), Amendment 1: MAC sublayer", 2012.
- [6] S. Petersen, et al., "WirelessHART Versus ISA100.11a: The Format War Hits the Factory Floor," *IEEE Industrial Electronics Magazine*, vol. 5, no. 4, pp. 23-34, Dec. 2011.
- [7] S. Montero, et al., "Impact of mobility on the management and performance of WirelessHART industrial communications," *Proc. 17th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Krakow (Poland), 17-21 Sept. 2012.
- [8] X. Zhu, et al., "RoamingHART: A Collaborative Localization System on WirelessHART," *Proc. 18th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pp. 241-250, Beijing (China), 17-19 Apr. 2012.
- [9] V. C. Gungor, et al., "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 4258-4265, Oct. 2009.
- [10] G. Sun, et al., "Time-Efficient Protocols for Neighbor Discovery in Wireless Ad Hoc Networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no 6, pp. 2780-2791, July 2013.
- [11] W. Sun et al., "Energy-Efficient Neighbor Discovery in Mobile Ad Hoc and Wireless Sensor Networks: A Survey," *IEEE Communications Surveys & Tutorials*, vol.16, no.3, pp.1448-1459, Third Quarter 2014.
- [12] A. A. Khan, et al., "Neighbor discovery in traditional wireless networks and cognitive radio networks: Basics, taxonomy, challenges and future research directions," *Journal of Network and Computer Applications*, vol. 52, pp 173-190, June 2015.
- [13] S. Vasudevan et al., "Efficient Algorithms for Neighbor Discovery in Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 21, no.1, pp. 69-83, Feb. 2013.
- [14] M. McGlynn, et al., "Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks," *Proc. 2nd ACM International Symposium on Mobile ad hoc Networking & Computing (MobiHoc)*, pp. 137-145, California (USA), 4-5 Oct. 2001.
- [15] T. Meng et al., "Code-Based Neighbor Discovery Protocols in Mobile Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 806-819, Apr. 2016.
- [16] H. Kaaranen, et al., "UMTS Networks: Architecture, Mobility and Services," *John Wiley & Sons*, 2001.
- [17] A. Kandhalu, et al., "U-connect: a low-latency energy-efficient asynchronous neighbor discovery protocol," *Proc. 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 350-361, Stockholm (Sweden), 12-16 Apr. 2010.
- [18] B. Zhang, Y. Li, D. Jin, and Z. Han, "Network science approach for device discovery in mobile device-to-device communications," *IEEE Transactions on Vehicular Technology*, vol. 65, no 7, pp. 5665-5679, July 2016.
- [19] V. Dyo et al., "Efficient Node Discovery in Mobile Wireless Sensor Networks," *Proc. 4th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pp. 478-485, Santorini Island (Greece), 11-14 June 2008.
- [20] S. Montero et al., "LAN-ND, a New Neighbour Discovery Protocol for Mobile WirelessHART Industrial Networks," *Proc. 18th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Cagliari (Italy), 10-13 Sept. 2013.
- [21] L. Chen, et al., "Never Live Without Neighbors: From Single- to Multi-Channel Neighbor Discovery for Mobile Sensing Applications," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 3148-3161, Oct. 2016.
- [22] M. Zhang, et al., "McDisc: A Reliable Neighbor Discovery Protocol in Low Duty Cycle and Multi-channel Wireless Networks," In *Proceedings of the 8th IEEE International Conference on Networking, Architecture, and Storage (NAS)*, pp. 1-9, Shaanxi (China), 17-19 July 2013.
- [23] IEEE 802.15.4-2006. Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications.
- [24] B. Li et al., "Incorporating Emergency Alarms in Reliable Wireless Process Control", *Proc. ACM/IEEE 6th International Conference on Cyber-Physical Systems (ICCPs)*, pp. 218-227 Seattle (USA), 14-16 Apr. 2015.
- [25] E. Tanghe, et al., "The industrial indoor channel: large-scale and temporal fading at 900, 2400, and 5200 MHz," *IEEE Transactions on Wireless Communications*, vol. 7, no. 7, pp. 2740-2751, July 2008.

[26] XDM2510H, 2.4 GHz IEEE 802.15.4 WirelessHART compliant radio module, RF Monolithics, Mar. 2011.

[27] Q. Dong, et al., "A Survey on Mobility and Mobility-Aware MAC Protocols in Wireless Sensor Networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 88-100, First Quarter 2013.