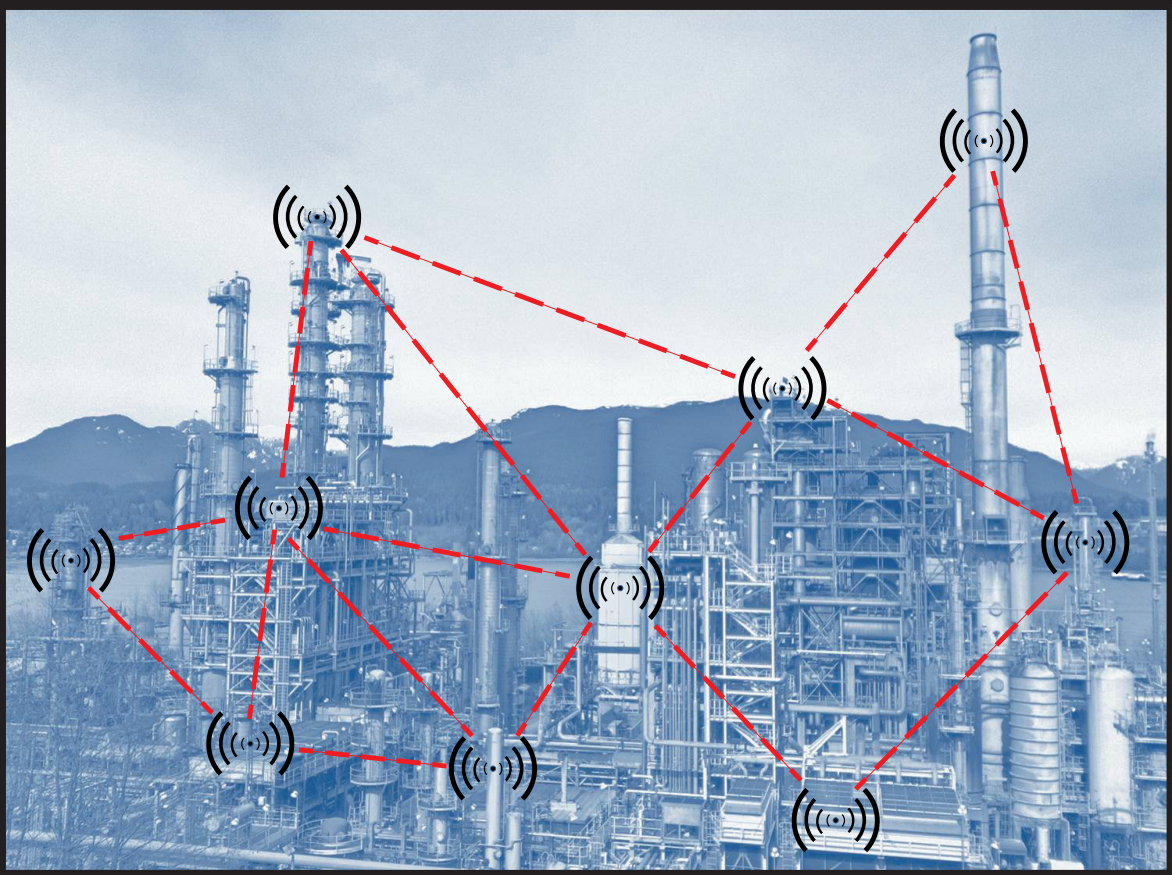
Decentralized management schemes for real-time and reliable communication in industrial wireless sensor and actuator networks



Pouria Zand

Decentralized management schemes for real-time and reliable communication in industrial wireless sensor and actuator networks

Pouria Zand

De promotiecommissie:

voorzitter en secretaris:	Prof. dr. ir. P.M.G Apers
promotor:	Prof. dr. ing. Paul J. M. Havinga
leden:	
Prof. dr. Boudewijn Haverkort	Universiteit Twente
Prof. dr. Sape Mullender	Universiteit Twente
Prof. dr. Kristofer S.J. Pister	University of Berkeley
Prof. dr. Thiemo Voigt	Uppsala University
Prof. dr. Antonio Liotta	Technische Universiteit Eindhoven

This research is supported, in part, by the EU FP7-ICT project WiBRATE (http://wibrate.eu), under the Grant No. 289041.

This research is also supported, in part, by EIT ICT Labs within the activity 'RICH - Reliable IP for Channel Hopping networks'.

CTIT

CTIT Ph.D.-thesis Series No. 14-320
Centre for Telematics and Information Technology
University of Twente
P.O. Box 217, NL – 7500 AE Enschede

ISSN 1381-3617
ISBN 978-90-365-3727-8
DOI: 10.3990/1.9789036537278
http://dx.doi.org/10.3990/1.9789036537278

Printed by Gildeprint Drukkerijen - Enschede
Cover design: Pouria Zand

DECENTRALIZED MANAGEMENT SCHEMES FOR REAL-TIME AND RELIABLE COMMUNICATION IN INDUSTRIAL WIRELESS SENSOR AND ACTUATOR NETWORKS

PROEFSCHRIFT

ter verkrijging van
de graad van doctor aan de Universiteit Twente,
op gezag van de rector magnificus,
Prof. dr. H. Brinksma,
volgens besluit van het College voor Promoties,
in het openbaar te verdedigen
op woensdag 27 augustus 2014 om 16.45 uur

door

Pouria Zand

geboren op 26 mei 1982
te Tehran, Iran

Dit proefschrift is goedgekeurd door:
Prof. dr. ing. Paul J. M. Havinga (promotor)

Acknowledgments

Time flies...this is the best description of how I feel after having spent the last four years working towards a PhD degree. Looking back, I am surprised and at the same time very grateful for all I have received through out my PhD life. For me, completing the PhD degree was probably the most challenging activity of the first 30 years of my life. Over the last four years, I have learned to be patient and never give up. If the question of "Do you do a PhD in your next life?" comes, my answer will be a definite "YES"!

First of all, I would like to express my utmost gratitude to my promotor/supervisor, Prof. Paul Havinga, for his prompt and useful advices during my research and for giving me the priceless freedom to choose my own subject, the one that I loved to work. All the discussions with him were very inspiring and fruitful. He has always been extremely supportive and approachable.

I would also like to thank Supriyo Chatterjea, Arta Dilo and Emi Mathews for guiding me and for the fruitful discussions we had during my PhD.

To all Pervasive Systems group members, I would like to leave here my gratitude for providing such a nice international working environment. I have learned uncountable new things during our PS discussion meetings, events, and lunch times.

Having contributed directly to my thesis, all the co-authors of papers published and used as basis for this thesis deserve a special acknowledgement: Supriyo Chatterjea, Kallol Das, Emi Mathews, Arta Dilo, and Jeroen Ketema.

I would like to thank Boudewijn Haverkort, Sape Mullender, Kristofer S.J. Pister, Thiemo Voigt and Antonio Liotta for accepting being part of my committee. I feel honored to have such experts in my defense.

To my friends, I'm grateful for the numerous good moments we shared during this time and I hope we can have many more in the near future.

To my parents-in-law, I would like to thank for supporting me during these four years and motivating me to pursue my PhD study.

To my family (my mother, my father and my sister), I would like to thank for their unconditional support in letting me pursue my study way even if this

means living a little further away from each other.

Finally, I want to thank my wife Marzieh. She has always supported me in every imaginable way. She's not just my wife but my best friend and I can't thank her enough for simply being the perfect companion one could ever wish for.

Pouria Zand
Enschede, August 2014

Abstract

Current wireless technologies for industrial applications, such as WirelessHART and ISA100.11a, use a centralized management approach in which a central network manager handles the requirements of the static network. However, such a centralized approach has several drawbacks. For example, it cannot cope with dynamicity/disturbance in large-scale networks in a real-time manner while it also incurs a high communication overhead and latency for exchanging management traffic.

In this thesis, we address the drawbacks of the centralized management approach utilized in WirelessHART and ISA100.11a for real-time industrial monitoring and control applications. More specifically, we propose new decentralized network management schemes to provide an end-to-end reliable and real-time communication for battery-powered and harvested-powered devices in a distributed manner. These schemes enable the network devices to join the network, schedule their communications, establish end-to-end connections by reserving communication resources to address real-time requirements, and cope with network dynamicity (e.g., node/edge failures) in a distributed manner.

To evaluate wireless protocols in the domain of industrial monitoring and control, a reference point is needed. To that end, we developed a WirelessHART simulator in NS-2 as a reference point to evaluate other protocols. We validated the WirelessHART simulator with a WirelesHART deployment at an industrial plant. To the best of our knowledge, this is the first implementation that supports the WirelessHART network manager as well as the whole stack of the WirelessHART standard.

To address the requirements of battery-powered I/O devices, we propose a distributed management scheme to address real-time and reliable communication requirements. This scheme considers the full mesh topology in which I/O devices are capable of participating in routing and distributed network management tasks, such as communication resources scheduling.

We then propose a second distributed management scheme for hybrid networks to be used for real-time industrial wireless automation. This scheme

addresses the requirements of energy constrained I/O devices. In this scheme, the I/O devices cannot participate in routing and distributed management tasks. The routers can dynamically reserve communication resources and manage the I/O devices in the local star sub-networks. We demonstrate that the proposed scheme achieves higher network management efficiency compared to the ISA100.11a standard, without compromising the latency and reliability requirements of industrial wireless networks.

To better support and address the requirements of energy harvested I/O devices, we extend ISA100.11a. The proposed extension makes management more decentralized by delegating a part of the management responsibility to the routers in the network. It also allows the I/O devices to choose the best routers according to different metrics using local statistics and advertised routers' ranks.

Samenvatting

De huidige draadloze technologieën voor industriële toepassingen, zoals WirelessHART en ISA100.11a, gebruiken een gecentraliseerde management aanpak, waarbij een centrale netwerk manager de eisen van het statische netwerk hanteert. Een dergelijke gecentraliseerde benadering kent verscheidene nadelen. Zo kan deze niet omgaan met dynamiek / verstoring in grootschalige netwerken op een real-time manier, terwijl hij ook een hoge communicatie-overhead en latentie creëert voor het uitwisselen van beheer verkeer.

In dit proefschrift richten we ons op de nadelen van de gecentraliseerde aanpak zoals die gebruikt wordt in WirelessHART en ISA100.11a voor real-time industriële monitoring en controle toepassingen. Meer specifiek stellen we nieuwe gedecentraliseerde netwerk management systemen voor om end-to-end betrouwbare en real-time communicatie voor batterij - aangedreven en energie opwekkende apparaten op een gedistribueerde manier aan te bieden. Deze regelingen stellen de netwerkapparaten in staat zich met het netwerkte te verbinden, hun communicatie te plannen, end -to-end -verbindingen tot stand te brengen door het reserveren van communicatie middelen om aan real-time vereisten te voldoen, en om op een gedistribueerde manier om te gaan met netwerk dynamica (bijv. knooppunt / edge falingen).

Om draadloze protocollen op het gebied van industriële monitoring -en controle te evalueren, is een referentiepunt nodig. Daartoe ontwikkelden we een WirelessHART simulator in NS–2, die als referentiepunt dient om andere protocollen te evalueren. We valideerden de WirelessHART simulator met een WirelesHART implementatie op een industriële installatie. Voor zover we weten, is dit de eerste implementatie die zowel de WirelessHART netbeheerder als de gehele stack van de WirelessHART-standaard ondersteunt.

Om aan de voorwaarden van batterij - aangedreven I/O- apparaten te kunnen voldoen, stellen we een gedistribueerd management plan voor dat voorziet in de eisen van real-time en betrouwbare communicatie. Dit plan beslaat de volledige mesh topologie waarin I/O- apparaten in staat zijn om deel te nemen aan routing en gedistribueerde netwerk management taken, zoals de planning

van communicatie middelen.

We stellen dan een tweede gedistribueerd management plan voor, te gebruiken voor real -time industriële draadloze automatisering in hybride netwerken. Deze regeling voorziet in de vereisten van energy constrained I/O-apparaten. In deze opzet, kunnen de I/O-apparaten niet deelnemen aan routing en gedistribueerde managementstaken. De routers kunnen communicatie middelen dynamisch reserveren en de I/O-apparaten in de lokale ster subnetwerken beheren. We laten zien dat de voorgestelde regeling een hogere netwerkmanagement efficiëntie behaalt dan de ISA100.11a standaard, zonder afbreuk te doen aan de latentie- en betrouwbaarheidseisen van industriële draadloze netwerken.

Om de eisen van energie geoogste I/O-apparaten verder te ondersteunen en te vervullen, breiden we ISA100.11a uit. De voorgestelde uitbreiding maakt het management meer gedecentraliseerd door een deel van de verantwoordelijkheid voor het beheer aan de routers in het netwerk te delegeren. Ook kunnen de I/O-apparaten de beste routers kiezen op basis van verschillende metrieken met behulp van lokale statistieken en de geadverteerde rangordes van de routers.

Contents

CHAPTER 1

Introduction

Present-day large-scale industrial monitoring and control systems may typically consist of thousands of sensors, controllers and actuators. In order to carry out their assigned tasks, it is essential for the devices to communicate. In the past, this communication was performed over point-to-point wired systems. Such systems, however, involved a huge amount of wiring which in turn introduced a large number of physical points of failure, such as connectors and wire harnesses, resulting in a highly unreliable system. These drawbacks resulted in the replacement of point-to-point systems with industrial computer networks known as fieldbuses. Over the past few decades, the industry has developed a myriad of fieldbus protocols (e.g. Foundation Fieldbus H1 [1], ControlNet [2], PROFIBUS [3], CAN [4], etc.). Compared to traditional point-to-point systems, fieldbuses allow higher reliability and visibility and also enable capabilities such as distributed control, diagnostics, safety, and device interoperability [5].

However, industrial processes are rapidly increasing in complexity in terms of factors such as scale, quality, inter-dependencies, and time and cost constraint. Similarly, the view of increasing complexity also holds when considering applications, which go beyond monitoring and also require control. Control operations have traditionally been carried out at the point of sensing, but more complex applications are now requiring distributed sensing and control. For example, in order to optimize overall energy usage, an industrial plant might require several pieces of machinery located in different parts of the plant to change their operational characteristics. This would require distributed sensing, control and subsequently actuation.

Wireless technologies have the potential to play a key role in industrial monitoring and control systems as they have certain key advantages over conventional wired networks. In addition to extensively reducing bulk and installation costs, the unobtrusiveness of the technology allows it to be deployed easily in areas which simply cannot be monitored using wired solutions (e.g.

in moving parts) [6]. Modifications of the network topology (in terms of the addition or reorganization of nodes) can also be easily carried out without incurring additional costs for wiring. Not being prone to damage due to corrosion or wear and tear, wireless systems also require less maintenance than their wired counterparts. Thus this unique combination of increased scalability and robustness through using distributed mechanisms makes wireless technologies an invaluable option for developing future industrial applications that require fine-grained, flexible, robust, low-cost and low-maintenance monitoring and control. However, wireless strategies also introduce a set of problems that can detrimentally affect various performance metrics. For example, the provision of real-time and reliable communication is an essential requirement for communication in harsh industrial environments in the presence of interference. The quality of a link between a source and destination node can heavily influence the success of the delivery of data to the destination. The challenges arise when delivering the sensor data toward the gateway or actuator in a harsh and dynamic industrial environment.

The main aim of this thesis is to design a network management scheme that fulfills the requirements of monitoring and process control applications. In the remainder of this chapter, we elaborate on the characteristics of wireless sensor and actuator networks in industrial automation and on the key points outlined in Section 1.1. Section 1.2, describes the application requirements for monitoring and process control applications in wireless industrial automation. The limitations of current technologies are discussed in Section 1.3. In Section 1.4, we discuss the research objective of this thesis and, linked to this, how our research question will be addressed. Next, we summarize the main contributions of this work in Section 1.5. Finally, an overview of the thesis is given in Section 1.6.

1.1 Industrial wireless sensor and actuator networks

Wireless industrial automation networks consist of *I/O devices* (sensors and actuators), *routers* and a *gateway* equipped with wireless devices. These are therefore the typical components that operate in each industrial wireless network. The I/O devices (or field devices) are sensors and actuators that are connected to the process and installed in the plant field. A router is a special type of device that does not possess a process sensor or control element and as such is not connected to the process itself. A gateway interconnects I/O devices with the plant automation system.

The primary goal of these industrial Wireless Sensor and Actuator Networks

(WSANs) is to perform monitoring and controlling tasks even in a harsh and dynamic industrial environment. Figure 1.1 shows how sensors and actuators can communicate with host applications through routers and gateways.

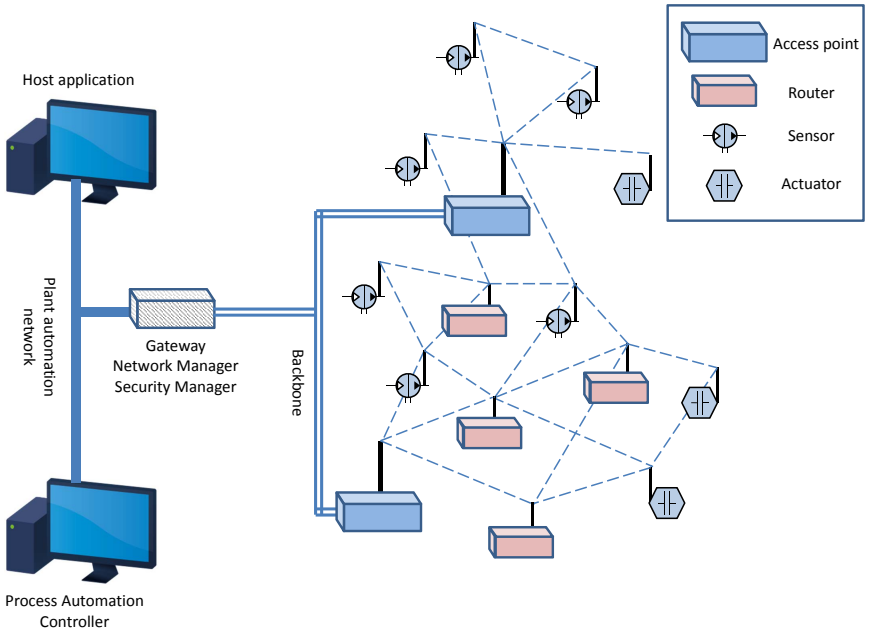


Figure 1.1: Example of wireless sensor and actuator network

1.1.1 Industrial WSANs Applications

Industrial control applications can be categorized into two main classes: (i) factory automation, and (ii) process control. Factory automation applications involve machines (e.g., robots) that perform discrete actions and are highly sensitive to message delays. Thus, such applications may require latency in the region of 2–50 ms. Process control, however, is typically used for monitoring and controlling the continuous production stream of fluid materials (e.g., oil and gas refinery) [7, 8]. Due to the non-critical nature of the process control applications, latency requirements are usually not stringent (>100 ms) [8].

Table 1.1: Different classes of applications as defined by ISA

Category	Class	Application	Description
Safety	0	Emergency action	Always critical
Control	1	Closed-loop regulatory control	Often critical
	2	Closed-loop supervisory control	Usually noncritical
	3	Open-loop control	Human in loop
Monitoring	4	Alerting	Short-term operational consequence
	5	Logging and downloading/uploading	No immediate operational consequence

ZigBee Pro, WSNs, WirelessHART, ISA100.11a

Increasing Importance of Message Timeliness

Based on the criticality and the importance of the applications, the International Society of Automation (ISA) considers six classes of applications, from critical control to monitoring applications, in which the importance of the message response time and Quality of Service (QoS) requirements vary [3]. In the more critical applications, sensor/process data need to be transmitted to the destination in a reliable, timely and accurate manner. Process control applications cover class 1 to 5 [7]. The details of the classes are shown in Table 1.1.

Traditional wireless sensor networks (WSNs) are deployed in class 4–5 applications, in which low-power consumption is given priority over the provision of a bounded response time delay. ZigBee Pro [9], one of the first standards for WSNs, is designed for applications which have soft real-time and reliability requirements. As a result, it can not address the requirements of industrial control applications [10, 11]. Similar to WSNs, ZigBee Pro is deployed in class 4–5 applications. ISA100.11a [12] and WirelessHART [13] standards are designed for process control and monitoring applications. ISA100.11a supports industrial applications from class 1 to 5, and WirelessHART supports industrial applications ranging from class 2 to 5 [8]. In this thesis we mainly focus on the (i) real-time and (ii) reliable communication requirements of periodic monitoring and process control applications from class 1 to 5 in industrial harsh and dynamic environments. In addition, we consider the requirements of harvested-powered I/O devices in dynamic environments.

Those applications generally involve unique characteristics that are discussed in the following sections.

1.1.2 Characteristics of WSANs

Wireless sensor and actuator networks have been designed to facilitate the implementation of a sensor and actuator communication system. In the remainder

of this section, we discuss the typical functionality of devices, different types of devices and the network scale in industrial wireless sensor and actuator networks.

1.1.2.1 Node functionality

In every WSANs-based solution, different kinds of tasks are required namely *routing*, *sensing/actuating*, *network managing* and *interconnecting the field devices with plant automation*. Additional details on the characteristics of these types of functionalities are given below:

1. ***Routing task***: the routing task is the process of forwarding data packets along the network toward the final destination between wireless nodes. Multiple routes can be constructed to allow for path diversity, depending on plant obstacles.

2. ***Sensing/actuating task***: the process of sensing includes measuring the physical environment. An actuation refers to a control process of a mechanism (or system) that involves movement.

3. ***Network management task***: network management is the process of forming a network, handling node affiliation, scheduling resources (e.g. defining superframes), configuring routing paths and monitoring and reporting network health. Network management can be classified into three classes namely *centralized*, *distributed* and *hybrid* management. In the centralized management approach a central network manager configures the networks. In contrast, in the distributed management scheme, the nodes participate in management tasks, such as communication schedule construction and routes establishment. The hybrid management scheme combines both approaches.

4. ***Interconnecting wireless and wired networks***: the nodes that are deployed in the plant, and which participate in wireless networks, need to be interconnected with the plant automation system. Various traffic flows need to be forwarded from the wired network to the wireless network and conversely.

1.1.2.2 Node classifications

In each WSANs-based solution, different kinds of devices (logical and/or physical) operate. These include *routers*, *I/O devices (or field devices)* , *access points*, a

gateway as well as a *network* and *security manager*. Additional details on these types of devices are given below:

1. ***Routers characteristics***: routers are deployed in the network to improve network coverage and connectivity. In WSANs, the routing role is usually executed by field devices. However, additional routers can be added to allow for path diversity, depending on plant obstacles. A router is a special type of field device that does not possess a process sensor or control element and as such is not connected to the process itself. A router may have the following additional characteristics, depending on the application requirements:

   - ***Management capabilities***: routers can be classified into routers with and without management capabilities. In some applications, routers with management capabilities use their own local resources to address the requirements of I/O devices and to allocate the requested bandwidth to them.
   - ***Router's rank***: in order to address the requirements of power-constrained I/O devices, the I/O devices need to know the ranks of the neighboring routers, to be able to dynamically choose the best possible neighboring router. Ranks are basically qualifying numbers defining the router's relative position/grade with respect to gateway(s). The routers advertise their ranks based on different Objective Functions (OFs) (e.g. reliability, latency, power consumption and available bandwidth). The rank may be calculated in either a distributed or in a centralized manner.

2. ***I/O device characteristics***: I/O devices (or field devices) are sensors and actuators that are connected to the process and installed in the plant field. The sensors are responsible for sensing (measuring) the physical environment. An actuator moves or controls a mechanism or system by functioning as a type of motor. In this thesis, we assume that the actuators support a set of function blocks for controlling purposes. The characteristics of I/O devices as outlined below may vary in various industrial automation applications. Hence, they affect the operation of the network.

   - ***Power Supply***: I/O devices generally contain batteries that provide energy to operate the wireless node. However, in some applications, I/O devices harvest energy from their environment. The resulting "fit-and-forget" technology that energy-harvested I/O devices introduce

is becoming particularly popular. The harvester-powered I/O devices might come with or without additional power sources. The availability of harvested energy typically varies over time in a non-deterministic manner. With today's energy harvesters, the I/O devices can perform only a few wireless transmissions/receptions per reporting cycle [14].

- ***Participation in routing and network management tasks***: the I/O device can participate in routing and network management tasks. The I/O device can perform distributed route construction and communication scheduling tasks. This depends on its memory/storage, processor and power supply. Should these resources be lacking, the I/O devices cannot perform routing and communication scheduling tasks.
- ***Mobility***: in most industrial applications, I/O devices are static devices. However, in some applications it is necessary that an I/O device be moved from one location to another. In that case, the I/O devices may be located on moving parts, such as rotating components, or be located on vehicles such as cranes or forklifts [7]. Furthermore, a wireless worker might need to be connected wirelessly and directly to the sensors and control points in or near the equipment on which he or she is working. In that case, the handheld device might be carried by the worker [7].

3. ***Access point***: access points are attached to the gateway and provide redundant paths between the wireless network and the gateway.

4. ***Gateway***: the gateway aims to interconnect field devices with the plant automation system by exploiting one or more access points. The gateway is responsible for data caching and query processing.

5. ***Network and security manager characteristics***: in the centralized management approach, the network manager aims to form a network, to handle node affiliation, to schedule resources (e.g. by defining superframes), to configure routing paths and to monitor and report network health. Redundancy is ensured thanks to the support of multiple (passive) network managers. The security manager handles security issues, e.g. by distributing encryption keys to the network manager of each network.

1.1.2.3 Network scale and topology

An industrial process control network lacks a specific physical topology, which can introduce challenges. Different use cases might require different types of

network topologies, such as star, linear, tree, mesh or the hybrid star-mesh. Also, the network-scale varies from one small hop to large scaled networks of several hops, according to the type of applications that are used.

However, in this thesis we mainly focus on those large scale networks that require reliable full mesh or hybrid star-mesh network topologies. This holds particularly true for a multi-square-kilometer refinery where isolated tanks, some of them equipped with power, but most with no backbone connectivity, compose a farm that spans the surface of the plant. In this environment, a few hundred I/O devices are deployed in a deterministic manner that need to be monitored and controlled. We therefore need to ensure global coverage using a wireless, self-forming, self-healing mesh network. The network size might be 5 to 10 hops. Powered infrastructure is typically not available in many parts of the network [7].

1.1.3 Traffic characteristics

1.1.3.1 Data model

The primary task of WSANs is to collect process data and send these in monitoring and process control applications to the gateway and/or actuators. Data reporting models can be categorized as either periodic or bursty data. In the following section, we explain the characteristics of these two models.

1. ***Time-driven (Periodic data)***: data that is generated periodically and has a well understood data bandwidth requirement, which is both deterministic and predictable. Timely delivery of such data is often the core function of a wireless sensor network. To that end, resources are assigned permanently to the network to ensure that the required bandwidth stays available. Buffered data usually has a short time to live, and the newer reading overwrites the previous [7].

2. ***Event-driven (Bursty data)***: this category includes alarms and aperiodic data reports with bursty data bandwidth requirements. In certain cases, alarms are critical and require a priority service (that would prioritize the message) from the network [7].

1.1.3.2 Traffic pattern

Three basic traffic flows should be supported by the WSANs. These traffic flows are: Point-to-Point (P2P), Multipoint-to-Point (MP2P), and Point-to-Multipoint

(P2MP) [15].

1. ***Point-to-Point traffic***: this traffic is usually between the I/O devices within the network. In this type of traffic, any node might communicate with any other node in the network.
2. ***Multipoint-to-Point traffic***: this traffic is usually from I/O devices inside the network towards a gateway (or network manager).
3. ***Point-to-Multipoint traffic***: this traffic is usually from a gateway (or network manager) to a subset of I/O devices inside the network.

1.1.3.3 Traffic rate

Most of the traffic in the network consists of real-time sensor data that is published periodically toward the other sensors, actuators or the gateway for closed-loop process control and monitoring applications. In general, the traffic rate and network throughput varies in different WSANs' use cases. However, in this thesis we mainly focus on those applications in which the rates vary from 1 per second to 1 per hour [7].

1.1.3.4 Message Priority or classification

The priority of MAC layer messages is dictated by their contents. Generally, there are four priority levels in industrial automation [13, 12]:

1. ***Management and network control messages (highest priority)***: any packet containing a payload with network-related diagnostics, critical management, configuration, or control information is classified with a priority of "Management" or "Network control".
2. ***Process/sensor data***: any packet containing process data and periodic real-time traffic shall be classified as priority level "Real-time Process-Data". This real-time process data is overwritten whenever a newer message is generated.
3. ***Sequential real-time data***: packets containing the low priority data that need sequential delivery of messages such as voice or video data.
4. ***Normal messages (lowest priority)***: MAC layer messages or client-server communications that do not meet the criteria for "Management", "Real-time Process-Data", or "Sequential Real-time Data" are classified as "Normal" priority.

1.2 Application requirements for industrial wireless solutions

Designing communication protocols for industrial WSANs is closely related to their application requirements. It is therefore impossible to design a single communication protocol that functions both effectively and efficiently for all kinds of WSNs applications. This section discusses the most essential metrics for large-scale industrial monitoring and control applications, such as real-time capability, scalability, power consumption and robustness.

- ***Real-time***: as discussed in Section 1.1.1, based on the criticality and importance of the applications, the International Society of Automation (ISA) considers six application classes, from critical control to monitoring applications, in which the importance of the message response time and Quality of Service (QoS) requirements varies [8]. In the more critical applications, process values need to be transmitted to the destination in a reliable, timely and accurate manner. The details of the classes are shown in Table 1.1.

  Certain Quality of Service (QoS) mechanisms are used by communication networks to meet the real-time requirements. These mechanisms can generally be categorized into: (i) traffic classification and (ii) resource reservation. The traffic classification mechanism can be used for channel access and packet delivery along the path between the endpoints, by labeling the packets with a priority value and placing them on the corresponding queue in the path. The resource reservation technique allocates the communication resources along the path between two end-points for a specific traffic or class of traffic to achieve the desired QoS requirement [16].

- ***Reliability***: reliability is an integral part of any industrial monitoring and control system as any slight degradation in communication can potentially result in complete system malfunction. In order to ensure reliable wireless communication, various techniques can be used to mitigate communication problems such as interference and weak signals. For example, channel hopping and multipath routing are suitable schemes to provide reliable communication by mitigating deep fading and external interference [17].

- ***Scalability***: as industrial processes increase in complexity, the number of points that need to be monitored and controlled increases rapidly. This makes it essential to design network architectures, which are capable of scaling up.

In other words, the objective is to ensure optimal network performance even when the network size or rate of data generation increases.

- ***Power Consumption***: process control is typically used for monitoring fluids (e.g., oil level in a tank, pressure of a gas, etc.). Such applications that typically involve non-critical applications requiring closed-loop control usually transmit process values at regular intervals. Furthermore, due to the non-critical nature of the process control applications, latency requirements are not usually stringent (>100 ms). This allows nodes to reduce power consumption by carrying out aggressive duty cycling of their radios and sensor sampling operations. In addition, in this class energy-harvested I/O devices with or without additional power sources are becoming popular.
- ***Management efficiency***: network management can be classified into (i) centralized, (ii) distributed, and (iii) hybrid management approaches. The management schemes might be more or less efficient depending on network conditions (e.g. static or dynamic). Issues such as node (re)joining, reserving communication resources, and the handling of network dynamicity (such as node or edge failures) will be affected by the management scheme that was selected.

Fulfilling the above mentioned requirements is challenging. Current wireless technologies fail to do so. The next section discusses their limitations.

1.3 Limitation of the current wireless technologies

Several wireless networking standards based on IEEE 802.15.4 [18], such as ZigBee Pro [9], WirelessHART [13] and ISA100.11a [12], are developed to support industrial applications. ZigBee Pro is not designed to support industrial process control applications, which have strict latency and reliability requirements [10]. WirelessHART and ISA100.11a are the two standards most widely accepted by the industry that use a centralized network management approach. While a centralized approach can generate optimal results for static networks, it has several drawbacks. Firstly, the network manager is prone to a single point of failure. In case of failure or network partitioning, nodes that do not have access to the network manager are left without management functionality. Secondly, the centralized approach incurs a high communication overhead and latency for exchanging management traffic. Thirdly, they cannot cope with network dynamicity in a timely manner. That is because the link quality between I/O

devices and routers may vary considerably due to the interferences in harsh industrial environments. Having the I/O devices rejoin the network and coping with such dynamic situations is costly, as several message exchanges are required to fix the broken links, which incurs high latency [19]. Additionally, the energy-harvested I/O devices might temporarily lose their power as well as their network connectivity, causing additional rejoining processes. These problems are exacerbated as the network scales up. We show in this thesis that these problems are significant and we demonstrate how they can be solved.

1.4 Research objective

This thesis aims to address the (i) real-time and (ii) reliable communication requirements of periodic monitoring and process control applications in industrial harsh and dynamic environments. It also seeks to explore how better efficiency in network management, in terms of delay and overhead issues, can be achieved. Although security is an important requirement, this subjects is beyond the scope of this thesis. Instead, we concentrate our efforts on scalable network management schemes that also address the high throughput requirement of some monitoring and process control applications. Furthermore, the requirements of battery-power and harvested-power I/O devices will be considered.

The main research question of this thesis is therefore:

> *How to provide a reliable and real-time communication network to address wireless automation requirements in a harsh and dynamic industrial environment, while achieving higher efficiency in network management in terms of delay and overhead?*

1.4.1 Hypotheses

In order to provide reliable and real-time end-to-end communication for (i) battery-powered and (ii) harvested-powered devices, we start from the hypothesis that various management schemes can be applied to manage industrial wireless networks. Generally, these network management schemes can be classified into (i) centralized, (ii) distributed and (iii) hybrid management approaches.

We consider the hypothesis that the distributed and hybrid management approach can easily adapt to dynamics in large-scale industrial wireless networks and improve the drawbacks of centralized management approach.

To address the requirements of battery-powered I/O devices, we consider the hypothesis that the I/O devices are capable of participating in routing and distributed network management tasks, such as communication resources scheduling. Such actions result in a full mesh network topology and a purely distributed management scheme.

To address the requirements of harvested-power I/O devices that are unable to participate in routing and distributed communication resources scheduling tasks, we consider the hypothesis that the routing devices can have additional management capabilities. The routing devices can dynamically reserve communication resources and manage I/O devices in the local star sub-networks. This will result in a hybrid network topology: a full mesh topology among the routers and a star network between the I/O devices and routers. The allocation of communication resources by the routers can be managed either (i) in a purely distributed manner or (ii) by the central network manager. These two policies result (i) in a distributed and (ii) a hybrid management approach, respectively.

1.4.2 Proposed Solutions

WirelessHART and ISA100.11a are the two standards that are most widely accepted by the industry. These two technologies use the centralized management approach. We evaluate the WirelessHART standard as a reference point for the centralized management approach to assess its efficiency in providing reliable and real-time communication in dynamic large-scale industrial networks. The outcomes of the WirelessHART evaluation also apply to ISA100.11a networks, due to the similarities in their lower layers and network management schemes. WirelessHART supports full mesh topologies, in which all nodes (routers and I/O devices) are considered to have routing capabilities. On the other hand, in the ISA100.11a network, I/O devices can be defined as nodes with or without routing capabilities, which results in a hybrid star-mesh topology.

In order to improve the drawbacks of the centralized management approach, we propose two distributed management schemes. The first one addresses real-time and reliable communication requirements. This scheme considers the full mesh topology in which I/O devices are capable of participating in routing and communication scheduling tasks.

The second one is a distributed management scheme that addresses the requirements of harvested-power I/O devices. It supports the hybrid star-mesh topology in which the routers are able to manage the I/O devices by forming local sub-networks. The I/O devices can dynamically choose the best possible

routers to cope with harsh and dynamic industrial environments in case of interference.

ISA100.11a* is a hybrid management scheme that is designed to support the power-harvested I/O devices' requirements. It supports the hybrid star-mesh topology. The central System Manager manages the communication among the routers in the mesh network. The routers with management capabilities manage a star sub-network, including the I/O devices.

1.5 Contributions

Following on from the earlier mentioned research question, the main contributions of this thesis can be listed as follows:

(Contribution 1) Implementation and validation of WirelessHART simulator in NS-2: in this contribution, we evaluate and implement a WirelessHART simulator. WirelessHART, was introduced to address industrial process automation and control requirements. We use this standard as a reference point to evaluate other wireless protocols in the domain of industrial monitoring and control. This makes it worthwhile to set up a reliable WirelessHART simulator to achieve that reference point in a relatively easy manner. Chapter 3 explains our implementation of WirelessHART in the NS-2 simulator. According to our knowledge, this is the first implementation that supports the WirelessHART network manager as well as the whole stack of the WirelessHART standard. It also explains our effort to validate the correctness of our implementation, namely through validation of the implementation of the WirelessHART stack protocol and of the Network Manager. We evaluate the performance of our implementation in terms of delay and communication load in the network. This implementation offers an alternative to expensive testbeds for testing WirelessHART. Different parts of this work appeared in the following papers [20, 21]:

- P. Zand, A. Dilo, and P. Havinga, "Implementation of WirelessHART in NS-2 simulator," in IEEE 17th Conference on Emerging Technologies Factory Automation (ETFA), 2012, pp. 1–8.
- P. Zand, E. Mathews, P. Havinga, S. Stojanovski, E. Sisinni, and P. Ferrari, "Implementation of wirelesshart in the ns-2 simulator and validation of its correctness," Sensors, vol. 14, no. 5, pp. 8633–8668, 2014.

(Contribution 2) A distributed network management scheme for real-time industrial wireless automation: In this contribution, we propose a distributed

network management scheme, D-MSR. This management scheme enables the network devices to join the network, schedule their communications, establish end-to-end connections by reserving the communication resources for addressing real-time requirements, and cope with network dynamicity (e.g., node/edge failures) in a distributed manner. We demonstrate via simulation that D-MSR can address real-time and reliable communication as well as the high throughput requirements of industrial automation wireless networks, while also achieving higher efficiency in network management than WirelessHART, in terms of delay and overhead. The results of this work appeared in the following papers [22, 19]:

- P. Zand, S. Chatterjea; J. Ketema; P. Havinga, "A Distributed Scheduling Algorithm for Real-Time (D-SAR) Industrial Wireless Sensor and Actuator Networks". In Proceedings of the 2012 IEEE 17th Conference on Emerging Technologies & Factory Automation (ETFA), Krakow, Poland, 17–21 September 2012; pp. 1–4.
- P. Zand, A. Dilo, and P. Havinga, "D-MSR: A distributed network management scheme for real-time monitoring and process control applications in wireless industrial automation," Sensors, vol. 13, no. 7, pp. 8239–8284, 2013.

(Contribution 3) A distributed management scheme for hybrid networks to provide real-time industrial wireless automation: in this contribution, we propose a distributed management scheme named D-MHR, which can address the requirements of energy constrained I/O devices. In D-MHR, the routers can dynamically reserve communication resources and manage the I/O devices in the local star sub-networks. We demonstrate that DMHR achieves higher network management efficiency compared to the ISA100.11a standard, without compromising the latency and reliability requirements of industrial wireless networks. This work has been accepted for publication in the following papers [23, 24]:

- P. Zand, K. Das, E. Mathews, and P. Havinga, “D-MHR: A Distributed Management Scheme for Hybrid Networks to Provide Real-time Industrial Wireless Automation,” WoWMoM 2014 [forthcoming].
- P. Zand, K. Das, E. Mathews, and P. Havinga, “A Distributed Management Scheme for supporting energy-harvested I/O devices,” ETFA 2014 [forthcoming].

(Contribution 4) ISA100.11a*: The ISA100.11a extension for supporting energy-harvested I/O devices: we propose an extension to ISA100.11a to better fulfill

the requirements of energy constrained I/O devices. The proposed extension makes the management more decentralized by delegating a part of the management responsibility to the routers in the network. It also allows the I/O devices to choose the best routers according to the desired metric, by using local statistics and advertised routers' ranks. We show that the proposed extension can better address the real-time and reliability requirements of industrial wireless networks than the traditional ISA100.11a standard. It can achieve higher network management efficiency in terms of reducing the delay and overhead of I/O devices than the ISA100.11a standard. This contribution has been accepted for publication in the following paper [25]:

- P. Zand, E. Mathews, K. Das, A. Dilo, and P. Havinga, "ISA100.11a*: The ISA100.11a extension for supporting energy-harvested I/O devices," WoWMoM 2014 [forthcoming].

1.6 Organization of the thesis

Figure 1.2 shows how the remainder of this thesis is organized. Chapter 2 provides an overview of the state-of-the-art of wireless technologies in industrial monitoring and control applications. It details the journey thus far and the road ahead. Chapter 3 describes in detail the implementation and validation of the WirelessHART simulator in NS-2 (which corresponds to Contribution 1). Chapter 4 describes the distributed network management scheme D-MSR (Contribution 2). Chapter 5 discusses the distributed management scheme D-MHR, which can address the requirements of energy constrained I/O devices (Contribution 3). In Chapter 6, we propose an extension to ISA100.11a to better address the requirements of energy constrained I/O devices (Contribution 4). Finally, Chapter 7 concludes this thesis with a summary and suggestions for future work.

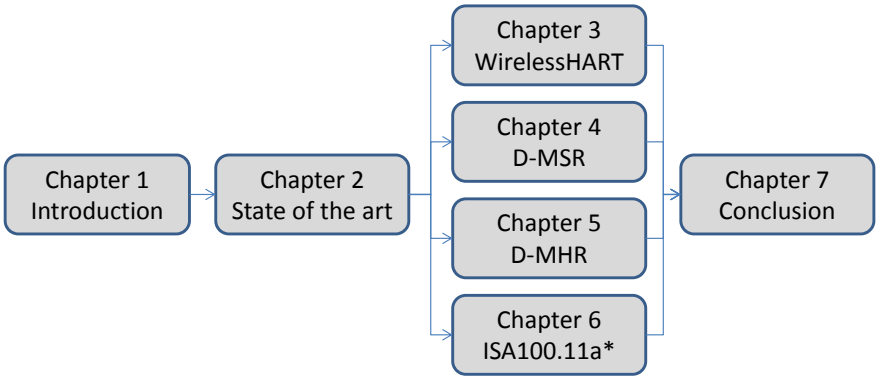


Figure 1.2: Organization of the thesis

CHAPTER 2

State of the art

While traditional wired communication technologies have played a crucial role in industrial monitoring and control networks over the past few decades, they are increasingly proving to be inadequate to meet the highly dynamic and stringent demands of today's industrial applications, primarily due to the very rigid nature of wired infrastructures. Wireless technology, however, through its increased pervasiveness, has the potential to revolutionize the industry, not only by mitigating the problems faced by wired solutions, but also by introducing a completely new class of applications. While present day wireless technologies made some preliminary inroads in the monitoring domain, they still have severe limitations especially when real-time, reliable distributed control operations are concerned. This chapter provides the reader with an overview of existing wireless technologies commonly used in the monitoring and control industry. It highlights the pros and cons of each technology and assesses the degree to which each technology is able to meet the stringent demands of industrial monitoring and control networks. The chapter also describes certain key research problems from the wireless networking perspective that have yet to be addressed to allow the successful use of wireless technologies in industrial monitoring and control networks.

2.1 Introduction

Present-day large-scale industrial monitoring and control systems may typically consist of thousands of sensors, controllers and actuators. In order to carry out their assigned tasks, it is essential for the devices to communicate. In the past, this communication was performed over point-to-point wired systems. Such systems, however, involved a huge amount of wiring which in turn introduced a large number of physical points of failure, such as connectors and wire harnesses, resulting in a highly unreliable system. These drawbacks resulted in the replacement of point-to-point systems using industrial computer networks known as *fieldbuses*. Over the past few decades, the industry has developed a myriad of fieldbus protocols (e.g., Foundation Fieldbus H1, ControlNet, PROFIBUS, CAN, etc.). Compared to traditional point-to-point systems, fieldbuses allow higher reliability and visibility and also enable capabilities, such as distributed control, diagnostics, safety, and device interoperability [5].

However, industrial processes are rapidly increasing in complexity in terms of factors such as scale, quality, inter-dependencies, and time and cost constraints. For example, globalization has led to companies opening up their manufacturing plants in not just one, but multiple geographic locations. Yet, in order to maximize the utilization of these distributed resources and optimize global operation, it is essential for companies to have a *detailed* outlook of the various operational characteristics of every single piece of equipment within every industrial plant. This could possibly require both static and moving parts of a piece of machinery to be monitored. In other words, accurate, fine-grained, large-scale, remote monitoring is an essential requirement [26].

Similarly, the view of increasing complexity also holds when considering applications which go beyond monitoring but also require *control*. Control operations have traditionally been carried out at the point of sensing, but more complex applications are now requiring distributed sensing and control. For example, in order to optimize overall energy usage, an industrial plant might require several pieces of machinery located in different parts of the plant to change their operational characteristics. This would require distributed sensing, control and subsequently actuation.

While existing industrial networking technologies are sufficient for performing localized monitoring and control, the distributed nature of upcoming industrial applications requires a paradigm shift from present-day strategies. The focus needs to shift from localized operations to a distributed approach where new benefits and synergies are discovered from the interconnection and communication of individual systems.

Wireless technologies have the potential to play a key role in industrial monitoring and control systems, as they have certain key advantages over conventional wired networks. In addition to extensively reducing bulk and installation costs, the unobtrusiveness of the technology allows it to be deployed easily in areas which simply cannot be monitored using wired solutions (e.g., in moving parts) [6]. Modifications of the network topology (in terms of the addition or reorganization of nodes) can also be easily performed without incurring additional costs for wiring. With increased scalability, wireless sensor networks can also run collaborative algorithms (e.g., for vibration monitoring applications) to improve the robustness of the overall system. Wireless systems also require less maintenance, since unlike their wired counterparts, they are not prone to damage due to corrosion or wear and tear. Thus, this unique combination of increased scalability and robustness through using distributed mechanisms makes wireless technologies an invaluable option for developing future industrial applications that require fine-grained, flexible, robust, low-cost and low-maintenance monitoring and control.

However, wireless strategies also introduce a set of problems that can detrimentally affect various performance metrics (e.g., reliability and real-time capability). In Section 2.2, this chapter provides the reader with an overview of existing wireless technologies commonly used in the monitoring and control industry. Section 2.3 highlights the pros and cons of each technology and assesses the degree to which each technology is able to meet the stringent demands of industrial monitoring and control networks. In Section 2.4 this chapter presents mechanisms used by industrial technologies for addressing the requirements of industrial automation wireless networks in terms of real-time capability and reliability. Section 2.5 describes key research problems from the wireless networking perspective that have yet to be addressed to allow wireless technologies to be successfully used in industrial monitoring and control applications. Finally, Section 2.6 concludes the chapter.

2.2 Overview of Existing Wireless Standards and Protocols

This section presents an overview of the wireless technologies that have been specifically tailored for use in industrial automation. They can be categorized into two parts, the IEEE 802.15.1 and IEEE 802.15.4 [18] based standards.

Wireless Interface for Sensor and Actuators (WISA) [27] is a protocol based

on the IEEE 802.15.1 standard. It has been developed by ABB and allows wireless communication between sensors and actuators. It is specifically designed to address the stringent real-time requirements of factory automation.

ZigBee Pro [9], WirelessHART [13], WIA-PA [28], ISA100.11a [12], and IEEE 802.15.4e [29] (Time Slotted Channel Hopping (TSCH) mode) are the IEEE 802.15.4 based standards. Among these, WirelessHART, WIA-PA, ISA100.11a and IEEE 802.15.4e are designed for industrial process automation requirements using concepts derived from the Time Synchronized Mesh Protocol (TSMP) [30] TSMP, developed by DustNetworks, is a media access and networking protocol that is designed for low power and low bandwidth reliable communication.

The WirelessHART protocol, developed by the HART Communication Foundation, uses a time-synchronized, self-organizing and self-healing mesh architecture. WirelessHART is backward compatible with the HART (Highway Addressable Remote Transducer) protocol, which is a global standard for sending and receiving digital information over analog wires between monitoring and control systems.

WIA-PA is a kind of system architecture and communication protocol of wireless networks that was first developed by the Chinese Industrial Wireless Alliance (CIWA).

ISA100.11a has been developed by the ISA100 standard committee, which is a part of the International Society of Automation (ISA). ISA100.11a uses IPv6 over Low power WPAN (6LoWPAN) protocol in the network layer. The 6LoWPAN was originally targeted at IEEE 802.15.4 radio standards assuming layer-2 mesh forwarding capability. Using the 6LoWPAN protocol in the network layer in ISA100.11a allows IP-based communication over IEEE 802.15.4. ISA100.11a uses a synchronized mesh protocol (based on TSMP) in the data link layer which allows peer-to-peer communication and mesh forwarding. This makes every node in the sensor network directly accessible through the Internet. WISA, WirelessHART, WIA-PA and Zigbee Pro do not have the capability to provide such access.

Our survey in [16] summarizes the main features of TSMP, IEEE 802.15.4e, WISA, ZigBee pro, WirelessHART, WIA-PA and ISA100.11a, as well as their main strengths and drawbacks.

2.3 Critical Metrics for Industrial Monitoring and Control

This section first evaluates the existing wireless technologies based on certain metrics that are essential for large-scale industrial monitoring and control applications, such as real-time capability, scalability, power consumption and robustness.

2.3.1 Real Time Capability

Based on the criticality and importance of the applications, the International Society of Automation (ISA) considers six classes of wireless communication, from critical control to monitoring applications, in which the importance of the message response time and Quality of Service (QoS) requirements varies [8]. In the more critical applications, process values need to be transmitted to the destination in a reliable, timely and accurate manner. The details of the classes are shown in Table 1.1

While ISA100.11a supports industrial applications from class 1 to 5, WirelessHART supports industrial applications ranging from class 2 to 5 [8]. ZigBee Pro is designed for applications which have softer real-time requirements [10]. Traditional wireless sensor networks (WSNs) are deployed in class 4–5 applications [8], where low-power consumption is given priority over providing a bounded response time delay. Such WSNs are not suitable for controlling tight control loops as nodes usually spend a large proportion of the time in a low-power sleep state.

WISA is the only wireless protocol that is suitable for factory automation applications as it can provide some strict real-time guarantees. There are related basic wireless requirements in such applications, for example, low additional latency due to wireless link (e.g., <10 ms).

We carry out a more detailed analysis of the real-time capabilities of ZigBee Pro, WirelessHART, WIA-PA, WISA and ISA100.11a later in the chapter by discussing specific details relating to the MAC layer contention mechanism and priority management schemes.

2.3.2 Scalability

As industrial processes increase in complexity, the number of points that need to be monitored and controlled increases rapidly. This makes it essential to

design network architectures which are capable of scaling up. In other words, the objective is to ensure optimal network performance even when the network size or rate of data generation increases.

Current wireless technologies designed specifically for industrial applications such as WirelessHART, WIA-PA (in centralized management scheme) and ISA100.11a mostly use a centralized approach for managing resources. While centralized approaches are technically easier to develop and manage, they are unable to cope with sudden changes that might occur frequently in a harsh industrial environment. This problem is further exacerbated as the network is scaled up. For example, a motor capable of running at different speeds may cause radio interference at different frequencies as it changes its operational speed. Wireless nodes operating in the vicinity of the motor should ideally reorganize their communication protocols using distributed techniques as and when interference is detected to quickly adapt to the changing environment. Traditional centralized approaches are unable to cope with such sudden unexpected changes, as they would then require detailed network statistics to be sent back to the central system manager which would then clog up the limited network resources. Thus, the larger the scale of the deployment, the more important it is to utilize distributed approaches to ensure that the system continues to perform optimally.

2.3.3 Power Consumption

Unlike traditional wireless sensor networks, power consumption has a lower priority than other performance metrics, such as reliability and real-time capability in industrial sensor networks. However, the degree of importance of power consumption varies greatly depending on the class of application. Industrial control applications can be categorized into two main classes: (i) process control, and (ii) factory automation.

Process control is typically used for monitoring fluids (e.g., oil level in a tank, pressure of a gas, etc.). Such applications which typically involve non-critical applications requiring closed-loop control usually transmit process values at regular intervals. Furthermore, due to the non-critical nature of the process control applications, latency requirements are not usually stringent (>100 ms). This allows nodes to reduce power consumption by carrying out aggressive duty cycling of their radios and sensor sampling operations. Factory automation applications, however, involve machines (e.g., robots) that perform discrete actions and are highly sensitive to message delays. Thus, such applications generate 'bursty' data and may require latency in the region of 2–50 ms. In

such instances, reducing power consumption has a lower priority than other performance metrics such as real-time capability and reliability.

In terms of energy consumption, ZigBee Pro and WIA-PA (in the cluster/star level) do not perform as well as the other competing technologies as it carries out time synchronization using the beacon-enabled mode of IEEE 802.15.4. Using beacons introduces a large overhead in terms of higher energy consumption, as the radio needs to remain in listen mode for long periods. Conversely, TSMP solves the time synchronization problem in a more energy-efficient manner by only relying on ACKs to exchange timing offset information. WirelessHART, WIA-PA (in the mesh level) and ISA100.11a also benefit from this approach as they both utilize TSMP. Furthermore, the ISA100.11a specification allows the transmission power of individual nodes to be controlled. This can result in additional energy savings. However, the specifications do not describe any algorithms indicating the strategies to be followed to carry out adaptive power control.

2.3.4 Reliability

Reliability is an integral part of any industrial monitoring and control system as any slight degradation in communication can potentially result in complete system malfunction. In order to ensure reliable wireless communication, various techniques can be used to mitigate communication problems such as interference and weak signals. Figure 2.1 gives an overview of the different classes of problems in wireless communication commonly present in industrial environments and their relevant solutions. We present some of the more important solutions developed in industry in greater detail in the following sections.

2.4 Mechanisms Used by Industrial Technologies to Improve Performance Metrics

This section discusses the mechanisms used by industrial technologies for addressing the requirements of industrial automation wireless networks in terms of real-time capability and reliability. The mechanisms include Media Access Control (MAC) layer contention techniques, priority management schemes, channel hopping, and multi-path routing.

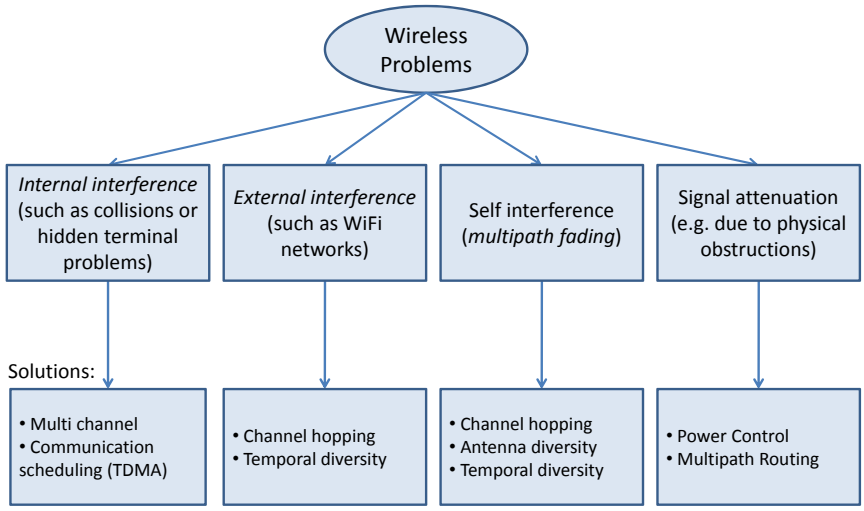


Figure 2.1: Common wireless communication problems and relevant solutions in typical industrial environments

2.4.1 MAC Layer Contention Mechanism and Communication Scheduling

A MAC protocol can generally be designed to operate using two mechanisms: (i) contention-free (scheduled communication) and (ii) contention-based. Contention-free approaches, e.g., dedicated timeslot-based, are more suitable for supporting real-time communication while shared timeslots (i.e., contention-based mechanisms) favor soft real-time applications.

Contention-based communication protocols, such as CSMA, are unable to provide timing guarantees when delivering messages. They are prone to packet loss by the hidden terminal problem (internal interference). Since ZigBee Pro runs on a CSMA-based MAC protocol, it is unsuitable for applications that require reliable and timely packet delivery, although WIA-PA and ISA100.11a use a CSMA-based MAC (slow hopping) for subnet discovery and retries. These latter are capable of switching to a slotted scheme where every link is scheduled to transmit at a predefined slot (TDMA) and channel offset, thereby avoiding the issue of internal interference. This mechanism is shown in Figure 2.2, in which the combination of slow hopping and slotted hopping is displayed. A

similar form of communication scheduling is also used in TSMP, WirelessHART, and IEEE 802.15.4e.

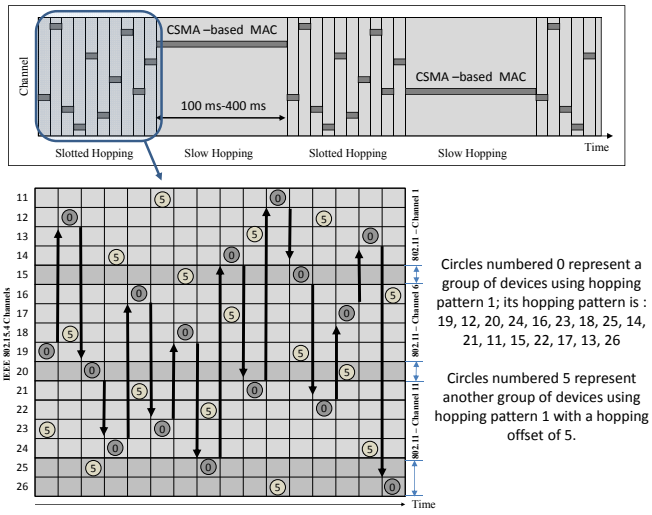


Figure 2.2: The ISA100.11a communication scheduling mechanism that alternates between a TDMA and CSMA-based scheme

Both WISA, WirelessHART, and WIA-PA use TDMA-based mechanisms with exclusively dedicated timeslots which do not support any variation in traffic [10]. However, ZigBee Pro, ISA100.11a, and the 802.15.4e MAC standard allow the user or centralized system manager to configure the timeslot length. This could be advantageous for coping with variable data traffic rates on the network which could be a characteristic of factory automation applications requiring real-time operations. However, as individual nodes are unable to make autonomous decisions, existing technologies are unable to provide hard real-time guarantees, especially in the presence of variable data traffic.

ISA100.11a and WirelessHART use a superframe management technique to maintain real-time communication for high traffic loads. The superframe period can determine several network performance parameters, such as packet delivery latency, energy consumption, and bandwidth utilization. ISA100.11a allows tradeoffs by enabling the system manager to determine the superframe period. A shorter-period superframe results in lower packet delivery latency

and higher bandwidth utilization, but results in greater energy consumption, while a longer-period superframe has the opposite effect. ZigBee Pro and WIA-PA (in the cluster/star level) also allow the transmission of superframes with different lengths in the beacon mode.

2.4.2 Resource reservation and traffic classification

Timely and reliable data transport is crucial for industrial automation applications. Communication networks are usually designed to meet such criteria by using certain Quality of Service (QoS) mechanisms. QoS mechanisms generally use two techniques to achieve their goals: (i) *traffic classification* and (ii) *resource reservation*.

The *traffic classification* mechanism can be used for channel access and packet delivery along the path between the endpoints, by labeling the packets with a priority value and placing them on the corresponding queue in the path. The *resource reservation* mechanism is used for allocating and reserving the resources along the path between two end-points for the specific traffic or class of traffic to achieve the desired QoS requirement.

Fox example, in the wired CAN protocol (a communication system for industrial and automotive applications), a MAC layer technique is used to resolve the contention between several nodes to access the channel. It involves bit-wise priority arbitration for collision resolution that relies on a node's ability to transmit and receive simultaneously. Each packet has a priority value that is used to resolve the contention among different nodes trying to access the channel. The node with a higher priority label in its data packet has a higher chance of accessing the channel. Each contender node transmits its priority value and receives feedback from the channel simultaneously. A node realizes that it has lost the contention when it detects a higher priority bit on the channel compared to the bit transmitted by the node itself [31, 32]

This technique cannot be used in wireless sensor networks as they typically have half-duplex transceivers. WIA-PA uses the traffic classification method for addressing different QoS requests. They define four priority levels, based on different classes of data: *command packets*, *process data*, *normal packets* and *alarm packets*. Low priority packets are declined when the device buffers become full. WirelessHART and ISA100.11a use a combination of traffic classification and resource reservation techniques for providing different QoS requests. When a device wants to establish communication with the central system manager or another device, it sends the contract request (Service request in WirelessHART), including input parameters, such as communication service type (scheduled or

unscheduled communication), destination address, traffic classification (best effort queued, real time sequential, real time buffer and network control), requested period, and committed burst for non-periodic communication, to the system manager. The system manager uses its centralized optimization algorithm to determine the required allocation of the network resources (such as graphs and links) and sends a contract response to the source after all necessary network resources have been configured and reserved along the path. However, WirelessHART and ISA100.11a do not specify the specific optimization algorithms that can be used by the system manager to allocate resources.

2.4.3 Channel Hopping Techniques

Channel hopping is often used to mitigate *external interference* and *multipath fading*. The proper reception of wireless signals may be prevented by other radio signals generated by the devices outside the network. This kind of interference is known as external interference. Signals in the same frequency range can be generated by Bluetooth devices, microwave ovens, other external networks (such as the IEEE 802.11 network) or many unintended sources of radio interference, such as other high-power interference sources. Channel hopping techniques are a way to mitigate external interference and multipath fading.

Figure 2.3 provides a classification of the different channel hopping techniques as well as the standards which use each of these techniques.

There is a tradeoff between using *blind channel hopping* and *adaptive channel hopping* (ACH). In the former, if the node switches to another congested channel or switches from a good channel to a congested one, this hopping does not help to mitigate the interference and just wastes energy [33]. However, in spite of this disadvantage, blind channel hopping has less overhead as the hopping pattern is already known by the network devices. In addition, if the system manager decides to blacklist a particular channel, nodes in the network still hop to the channel, but simply remain idle in that time period. Thus the larger the number of blacklisted channels, the more time is lost by nodes idling on blacklisted channels. Blind channel hopping techniques ensure that while two communicating nodes hop in unison, neighboring node pairs never use the same frequency at the same time in order to prevent hidden terminal problems. This is shown in Figure 2.4.

ACH differs from blind hopping in the sense that unlike in blind hopping, nodes do not keep on changing from one operating frequency to another at regular time intervals. In other words, nodes only change their frequencies when interference is detected on the current operating channel. However,

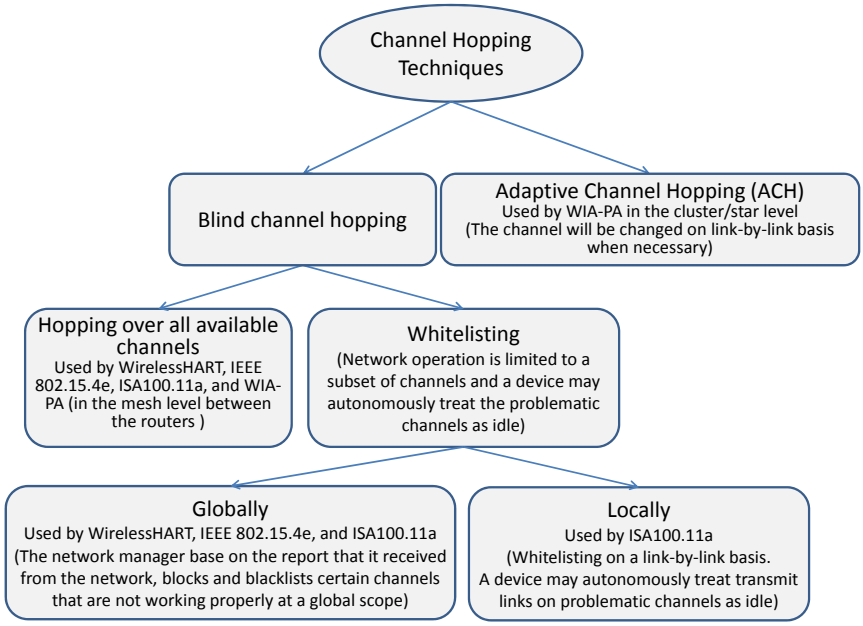


Figure 2.3: Channel hopping techniques

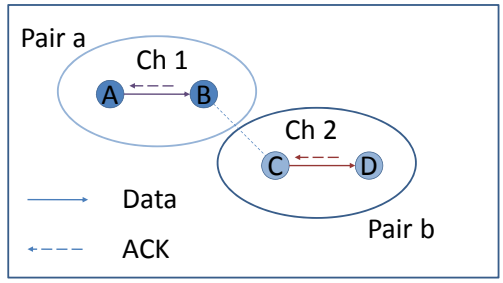


Figure 2.4: Two pairs of nodes using different communication channels

nodes need to collaborate to decide which channel to switch to and this can introduce a significant overhead since nodes need to continuously scan all

channels for interference levels and also because nodes need to ensure that while communicating nodes choose the same frequency, neighboring node pairs use different channels [34]. In WIA-PA in each cluster/star network, the cluster head and each node irregularly change their channel on a link-by-link basis only when channel conditions require it to. However to the best of our knowledge, there are currently no algorithms indicating strategies to be followed for ACH approaches in the multi-hop network.

Both WirelessHART and ISA100.11a networks use blacklisting techniques to mitigate external interference or multipath fading. WISA and ZigBee Pro do not have this capability.

ZigBee Pro, however, uses "frequency agility". This mechanism is not as tolerant to fluctuating wireless conditions as WirelessHART, WIA-PA, and ISA100.11a. In this technique, the network channel manager collects interference reports from all the nodes. If external interference is detected, the network channel manager scans for a better channel and moves the *entire* network to a new channel. This technique requires network formation to be carried out again and thus introduces inconvenient delays. Note that ACH is clearly a better technique as it only requires nodes facing interference to make changes to their operating frequencies—it does not affect the entire network.

ISA100.11a tries to separate the successive channels in the hopping pattern by at least 20 MHz. That means that at least a three-channel separation exists in each consecutive hop and, in the case of retries in the next hop, that they will not encounter the same IEEE 802.11 channel. The way by which this standard can coexist with the IEEE 802.11 standard has been predefined. This hopping separation is more than the *coherence bandwidth value*—the frequency-shift by which a link has to undergo transition from deep fade-in the case of indoor multi-path interference [17].

WISA employs frequency-hopping sequences in which the consecutive hops are widely separated in frequency and the sub-band bandwidth is more than the typical bandwidth of the coherence bandwidth. WISA uses frame-by-frame frequency hopping in which the used radio channel for retransmission is independent of the previous one, so likelihood of a successful transmission increases.

2.4.4 Multipath Routing

To ensure real-time capability, WISA protocols require the network to be deployed in a star topology rather than a multi-hop meshed network that is used in ZigBee Pro, WirelessHART, WIA-PA and ISA100.11a. A disadvantage of this approach is that if communication between a node and its cluster head

is disrupted due to interference, alternative routes cannot be established to transport the data. The multi-hop approach of ZigBee Pro, WirelessHART and ISA100.11a prevent such problems from occurring. Additionally, to ensure robust communication in WirelessHART and ISA100.11a, the system manager defines multiple paths for each node to reach a particular destination in the network.

2.5 Open Research Areas

While existing wireless technologies developed for industrial applications are able to carry out monitoring tasks fairly well, significant advances are required before they can be used for reliable, real-time, distributed control operations. We now highlight some of the key areas which need to be addressed to make this a reality.

2.5.1 A Distributed Approach to Achieving Real-Time Operation

Large-scale, distributed, real-time control applications require data to be transmitted over long distances through a multi-hop network in a timely manner. A distributed resource reservation algorithm is needed which would allow source nodes, based on the requirements of the application and traffic characteristic, to reserve network resources for its peer communications along their paths for addressing different QoS needs. The distributed nature allows the system to adapt quickly to disturbances or changes within the network to meet timing guarantees to support real-time control operation. While such mechanisms do not exist for present day sensor nodes in a distributed manner, relevant techniques from other networking-related domains could potentially be adapted to develop solutions that are suitable for wireless sensor and actuator networks. We briefly describe some of these relevant techniques.

QoS in multi-hop networks can be supported by different mechanisms, such as *circuit switching*, *Asynchronous Transfer Mode (ATM) networks*, and *internet protocols* (such as Integrated Services/RSVP, Differentiated Services, MPLS and constraint-based routing). It is also supported by IEEE 802.11e [35], ISA100.11a and WirelessHART in a centralized manner.

ATM [36] signaling protocols address certain performance issues in terms of reliability and timeliness of packet delivery that are of importance in industrial applications that require closed-loop, real-time control. The ATM protocol

uses a switching technique that combines the concepts of circuit switching and packet switching. For example, similar to circuit switching, before initiating data transfer, a virtual circuit is first established between the source and destination. The protocol also includes admission control mechanisms that help determine whether the required QoS guarantees can be provided. ATM uses statistical multiplexing techniques, similar to those used in packet switching, in order to cope with variable bit rates (i.e., "bursty" traffic). As a response, Chapter 4 and Chapter 5 introduce D-SAR signaling protocol [22]: a Distributed Scheduling Algorithm for Real-time applications based on concepts derived from Asynchronous Transfer Mode (ATM) networks [36]. The D-SAR protocol is used to establish an end-to-end connection and to reserve communication resources based on the traffic characteristics requested by the source node, along the path toward the destination. These traffic flows can be either a type of network management traffic (e.g. network layer control messages) or sensor data traffic that is published periodically by the sensor nodes toward actuators or gateway.

Internet protocols are mainly designed for multimedia applications. In those protocols, some mechanisms exist that allow a data receiver to request a special end-to-end quality of service for its data flows or classes of data. RSVP signaling is used by several internet protocols, such as *Integrated Services Architecture*, differentiated service, and MPLS, through which the application can reserve the resource and set up the path between the source and destinations.

The IEEE 802.11e, based on *traffic classification* mechanisms, provides different degrees of satisfaction for the users of the service. They define different priorities through which traffic can be delivered in several access categories. This differentiation is achieved by considering different amounts of time for sensing the channel to be idle and by considering different lengths of the contention window during backoff. This implies that high-priority traffic can access the channel by shorter back-offs than low-priority traffic. In addition, the packets are labeled with priority value and introduced into the corresponding queue in the path. Admission control in this standard as an important component limits the amount of traffic admitted into a particular service class so that the network resources can be efficiently utilized.

2.5.2 Distributed Network Management

WirelessHART, WIA-PA (in the mesh level) and ISA100.11a use centralized network management techniques for communication scheduling and managing routes. While such an approach may be easier in terms of implementation, they

have numerous disadvantages. Centralized systems often perform poorly in terms of reaction time, as all updates need to be sent first to the centralized system manager (i.e., gateway) for further processing. The network/system manager then performs recalculations and disseminates updated instructions to the relevant nodes in the network. As the round-trip time for such decision-making actions can be very high (especially when network contention is high), centralized approaches are unable to cope with highly dynamic situations (e.g., "bursty" data traffic/varying link quality, and node mobility). This problem is further exacerbated as the network is scaled up. This in turn may result in problems, including increased packet loss and delayed data delivery, which increase energy consumption. The distributed nature of a distributed approach allows the system to adapt quickly to disturbances or changes within the network in real-time. However, current wireless control technologies that use distributed approaches also perform poorly in terms of reliability, efficiency and robustness. Chapter 4 proposes in response a distributed network management scheme, D-MSR. In addition, Chapter 5 discusses a distributed management scheme named D-MHR, which addresses the requirements of energy constrained I/O devices. Finally, in Chapter 6, we propose an extension to ISA100.11a to better address the requirements of the energy constrained I/O devices. This extension uses a hybrid (centralized and distributed) network management scheme.

2.5.3 Distributed or Centralized Radio Transmission Power Control

The transmission power used by a node can have a direct impact on the radio link quality, the level of interference and energy consumption. Ideally, all nodes should always use the least transmission power that will allow them to carry out their assigned tasks effectively. While the ISA100.11a specification allows the transmission power of individual nodes to be controlled, it does not describe any specific algorithms to perform adaptive power control. Several autonomous power control strategies, developed specifically for WSNs, can be found in the literature [37, 38]. However, they all have certain drawbacks which would prevent them from being used in a harsh industrial environment. For example, while the technique presented in [37] can adapt, it is not designed to handle rapid link quality fluctuations that could be caused by moving metal objects or electromagnetic interference from motors or pumps that may be common in an industrial environment. While the authors in [38] rightly point out that interference is an issue that needs to be addressed when developing adaptive

power control algorithms, the presented solution does not perform optimally as it is unable to correctly distinguish between weak signals and interference. This is an area that still requires further investigation.

2.5.4 Network Management Algorithms for Different Traffic Patterns

WirelessHART, WIA-PA (in the mesh level) and ISA100.11a use centralized network management techniques for communication scheduling and managing routes. However, those standards do not specify the specific optimization algorithms that can be used by the system manager to allocate resources. In [39], [40], [41], [42], [43], [44], and [45] the authors have proposed the centralized scheduling algorithm in WirelessHART for convergecast by considering linear, tree and mesh networks models. ISA100.11a standard supports peer-to-peer communication, in addition to uplink and downlink traffics. This feature makes the communication scheduling and route managing algorithm more complicated than WirelessHART, in which the main concern is forwarding the traffic toward the gateway and vice versa. To the best of our knowledge, there are currently no algorithms indicating strategies to be followed for communication scheduling and route formation in ISA100.11a.

2.6 Conclusions

Traditional wired industrial networking technologies have numerous drawbacks. They lack flexibility, face reliability issues (due to wear and tear) and are expensive to deploy and maintain. Wireless technology, however, through its increased pervasiveness, can introduce a completely new range of industrial applications as it has the potential to provide fine-grained, flexible, robust, low-cost and low-maintenance monitoring and control. While present-day wireless technologies have taken a step in the right direction, they still have severe limitations, especially when real-time, reliable distributed control operations are concerned. This chapter presented an overview of current wireless technologies and their deficiencies, and described some key research issues that still need to be addressed in order to successfully extend the use of wireless technologies to the industrial monitoring and control sector.

CHAPTER 3

Implementation of WirelessHART in NS-2 simulator and validation of its correctness

One of the first standards in the wireless sensor networks domain, WirelessHART, was introduced to address industrial process automation and control requirements. This standard can be used as a reference point to evaluate other wireless protocols in the domain of industrial monitoring and control. This makes it worthwhile to set up a reliable WirelessHART simulator in order to achieve that reference point in a relatively easy manner. Moreover, it offers an alternative to expensive testbeds for testing and evaluating the performance of WirelessHART. This chapter explains our implementation of WirelessHART in the NS-2 network simulator. According to our knowledge, this is the first implementation that supports the WirelessHART Network Manager as well as the whole stack (all OSI layers) of the WirelessHART standard. It also explains our effort to validate the correctness of our implementation, namely through the validation of the implementation of the WirelessHART stack protocol and of the Network Manager. We use sniffed traffic from a real WirelessHART testbed installed in the Idrolab plant for these validations. This confirms the validity of our simulator. Empirical analysis shows that the simulated results are nearly comparable to the results obtained from real networks. We also demonstrate the versatility and usability of our implementation by providing some further evaluation results in diverse scenarios. For example, we evaluate the performance of the WirelessHART network by applying incremental interference in a multi-hop network.

3.1 Introduction

Despite the advancement of the realm of Wireless Sensor Networks, their adoption by the industry for factory automation and process control applications remained limited. This all changed when in 2007 the HART Communication Foundation [13] developed WirelessHART, the first open, international standard to fulfill industrial requirements. Using a self-organizing and self-healing mesh network architecture, it establishes a secure and reliable wireless communication protocol. It is backward compatible with the widely-used wired HART (Highway Addressable Remote Transducer) protocol: the global standard for sending and receiving digital information over analogue wires between monitoring and control systems. The WirelessHART standard has gained the confidence of the industry and it has been increasingly adopted over the last few years [46].

The International Society of Automation (ISA) considers six classes of applications, from critical control to monitoring, in which the importance of the message timeliness and Quality of Service (QoS) requirements decreases from class 0 to 5 in the Table 1.1 [8]. WirelessHART supports industrial applications ranging from class 2 to 5 [8].

Being the first open standard, WirelessHART can be used as a reference point to evaluate other wireless protocols in the industrial domain. This can be conveniently achieved by implementing the WirelessHART protocol in a network simulator. In addition, such implementations serve as a basis for further extensions and improvements of the protocol itself. Furthermore, to test and analyze the protocol easily, simulation provides a good alternative to expensive testbeds that need to be setup in real industrial environments. These factors motive us to work on implementing the WirelessHART simulator protocol. To that end, we choose one of the most popular network simulators, NS-2 [47] for our implementation.

Although WirelessHART has been partially implemented in other simulators [48], to the extent of our knowledge this is the first and complete WirelessHART simulator. This means that in our simulator, we implement the entire WirelessHART stack (all OSI layers) of field devices and access points and also the algorithms for centralized network management. A preliminary version of the simulator has been discussed in [20]. In this chapter, we present the implementation of the WirelessHART simulator, which adds a security layer to provide secure and reliable communications. In addition, we validate the simulator by using sniffed/captured traffic from a real WirelessHART network.

The rest of the chapter is organized as follows: Section 3.2 provides back-

ground information on the concepts used in WirelessHART and summarizes related works. Section 3.3 explains the WirelessHART architecture while Section 3.4 provides the implementation details of the WirelessHART device stack and the WirelessHART central network management algorithm. Methods on validating the simulator are discussed in Section 3.5. Experimental analysis of the real and simulated networks demonstrating the similarities and differences of network management algorithms is given in Section 3.6. Additional experiments in a multi-hop simulated scenario demonstrating the usability of the simulator is described in Section 3.7. Section 3.8 describes how the simulation tools can be used and finally, Section 3.9 concludes the chapter.

3.2 Background and Related Work

In any industrial network, the major concern is to provide real-time and reliable communications. Resource reservation is one of the techniques that can facilitate real-time communication. Channel hopping and multipath routing are two suitable schemes to provide reliable communication by mitigating the deep fading and external interference. These schemes were first proposed in the Time Synchronized Mesh Protocol (TSMP) [49] and were later adopted in the WirelessHART standard. In this section, we provide some background information on TSMP and provide a summary of relevant works on WirlessHART simulation.

3.2.1 Time Synchronized Mesh Protocol (TSMP)

TSMP is the first medium access and networking protocol designed for low power - low bandwidth reliable communication that utilizes all of the above mentioned techniques. TSMP concepts are used in several existing industrial wireless technologies such as WirelessHART, ISA100.11a and IEEE 802.15.4e (TSCH mode). IEEE 802.15.4e TSCH mode is a MAC amendment of the 802.15.4-2006 standard to support the industrial applications. TSCH is based on a time-slotted mechanism, where a schedule dictates on what slot and which channel a node should transmit/receive data to/from a particular neighbor.

TSMP divides the wireless channel into time and frequency. Time is divided into superframes, which consist of a collection of discrete time slots. Figure 3.1 illustrates the TSMP matrix for a sample network with a superframe of 10 slots. A single element in the TSMP superframe is called a cell. A link is a transaction that occurs within a cell. Link information consists of a superframe ID, source

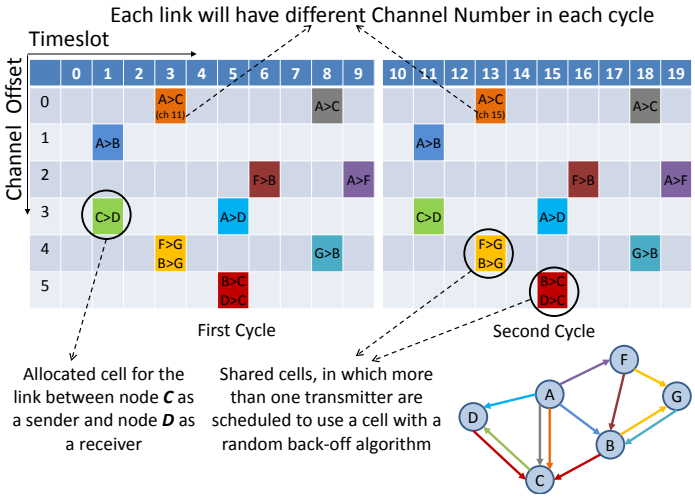


Figure 3.1: TSMP Slot-channel matrix for a sample network

and destination IDs, a slot number referring to the beginning of the superframe, and a channel offset. The two nodes at either end of the link communicate periodically once in every superframe. If only one transmitter is scheduled, the cell is contention-free. If multiple transmitters are scheduled for transmitting to the same device in a shared cell simultaneously, a random back-off algorithm can be used. Multiple links can be allocated from one node to another in different cells. For example, two "Txlinks" from node A to C are shown in Figure 3.1. TSMP links hop pseudo-randomly over a set of predefined channels. The radio channel used for communication is determined by considering the *timeslot number (ASN)*, *channel offset* and *channel hopping sequence* that can be formulated as follows:

$$Actual\ Ch\ \# = ChannelHoppingSequence((ASN + ChannelOffset) \\ \%Number\ of\ Channels) \tag{3.1}$$

Figure 3.2 depicts the specific timing requirement inside a TSMP timeslot. The scheduled communication in a timeslot between two nodes relies on accurate time synchronization across the network. The network devices should have the same notion of when each timeslot begins and ends. TSMP, unlike the IEEE 802.15.4 that uses the beacon-based synchronization scheme, relies on

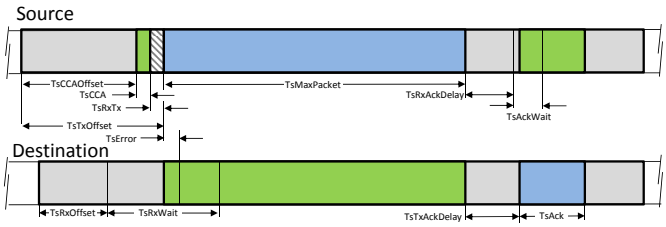


Figure 3.2: Timing of a dedicated TSMP timeslot

exchanging timing offset information of the received and sent packets to provide synchronization. The mechanisms for time synchronization are described in [13].

Graph table on node 13

Graph ID	Neighbor address
1	22, 23
2	23, 24

Graph table on node 5

Graph ID	Neighbor address
1	12, 13
2	13, 14

Graph 1 (To node 43)

Graph 2 (To node 45)

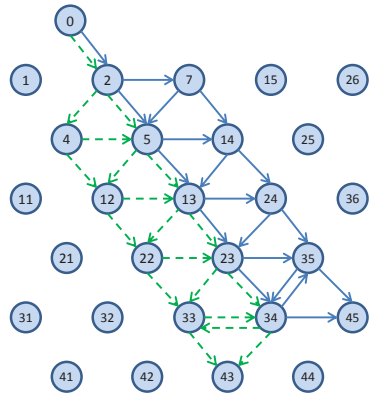


Figure 3.3: Graph routing sample

TSMP works on graph routing-based schemes. A graph is a routing structure that establishes directed end-to-end connection among devices. Each destination has its own graph, and several sources can share the same graph. Each graph in a network is identified with a unique Graph ID. Figure 3.3 illustrates the graph routing. In this figure, node 0 uses graphs with IDs 1 and 2 to communicate with nodes 43 and 45 respectively. When a source node wants to send a packet to a destination, the Graph ID will be included in the packet header to enable

routing to the destination. At any node in the path, multiple next hops could be specified in a mesh graph; path diversity is directly built-in [49]. In Figure 3.3, for example, an intermediate node 5 may forward a packet identified with Graph ID 1 to node 12 or node 13 and may forward a packet identified with Graph ID 2 to node 13 or node 14.

3.2.2 Related Work

Existing implementations of WirelessHART are partial. Nobre et al. [48] have developed a WirelessHART module for the NS-3 simulator. The focus of that work was on implementing the Physical layer of WirelessHART to use it as a basis for developing other layers, such as the Medium Access Control (MAC) and Application layer. In [50], the authors report on the development of the Physical and MAC layer of WirelessHART in OMNET++ [51]. This tool analyzes the effect of interference on the WirelessHART network. However, they did not implement the full WirelessHART stack nor the network management algorithms. In [52], the authors did implement a WirelessHART simulator based on TrueTime, an open source Matlab/Simulink-based tool for simulating networks, to study the clock drift in process control. However, in that research, the WirelessHART management algorithm and the whole stack were not implemented either. Shah et al. [53] implemented WirelessHART based on their previous work in TrueTime [52] and they abstract away from the Physical layer of the communication and move toward the application levels and control loops. They did not, however, cover multi-hop and multi-channel communication. The authors in [54] propose the use of a co-simulation framework based on the interaction of TrueTime, together with a cross layer wireless network simulator based on OMNET++ for improving overall coexistence management.

3.3 WirelessHART architecture

The WirelessHART protocol has been designed in order to implement a sensor and actuator mesh communication system. A typical topology of a WirelessHART network showing its architecture is depicted in Figure 3.4. The following types of devices (logical and or physical) operate in the network:

- Security Manager (SM), whose task is to handle security issues, e.g. the distribution of encryption keys to the Network Manager in each network.

- Network Manager (NM) per network, which forms the network, handles node affiliation, schedules resources (e.g. defining superframes), configures routing paths, monitors and reports the network health etc. Redundancy can be ensured by using multiple (passive) NMs.
- Gateway (GW), whose task is to interconnect field devices with the plant automation system by exploiting one or more access points.
- Access points are attached to the gateway and provide redundant paths between the wireless network and the gateway.
- Routers are deployed in the network to improve network coverage and connectivity. In WirelessHART, the routing role is usually executed by field devices. However, additional routers can be added to allow for path diversity, depending on plant obstacles. A router is a special type of device that does not possess a process sensor or control element and as such is not connected to the process itself.
- Several field devices, i.e. sensors and actuators, that are connected to the process. All these devices are able to participate in routing tasks.

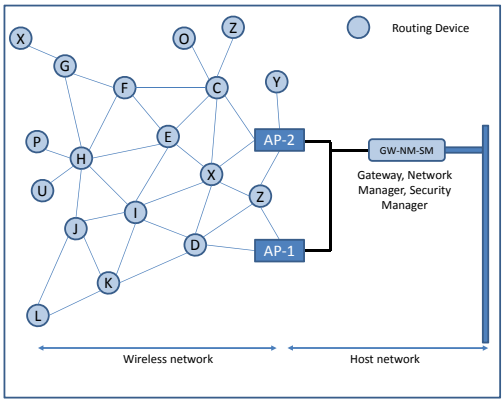


Figure 3.4: A sample WirelessHART network topology

In addition, there are also other devices with wireless communication interfaces, but those are not connected to the process and are installed in the

plant field. Examples include handheld terminals used for commissioning and maintenance purposes and so-called "adapters" that connect legacy hardware with the wireless network.

Commercially available devices often embed the GW-NM-SM roles into a single physical device as shown in Figure 3.4. Such a centralized approach allows all the computational burdens to be confined to a single device, thereby reducing the costs of field devices. All communication occurs, by moving data to/from the gateway, through the intermediate routing devices, thereby following the preassigned routing path. This architecture, despite its simplicity, ensures efficiency in a plant network in which nodes are rarely reconfigured or added during the network's lifetime and where network requirements are rather static.

Furthermore, a centralized architecture facilitates the implementation of a wide variety of network topologies, e.g. according to peculiar application requirements. In a high-performance scenario, it is probably better to adopt a star topology (i.e. all devices are one hop away from the gateway). In contrast, a multi-hop mesh topology is useful for a less demanding scenario (from the timings point of view) like monitoring. Any type of intermediate topology, e.g. cluster-tree networks, can also be realized.

3.4 WirelessHART Implementation

As existing implementations of the WirelessHART are rather incomplete, we decided to implement a complete implementation, including the WirelessHART stack as well as the GW-NM-SM functionalities. The management algorithm described in [55] was selected for the NM. It is one of the few network management algorithms that addresses both routing and communication scheduling.

3.4.1 WirelessHART protocol stack

The WirelessHART protocol stack is shown in Figure 3.5. All field devices and access points in the network should support this stack.

3.4.1.1 Physical layer

The Physical layer of WirelessHART is the IEEE 802.15.4 standard's Physical layer, which already exists in the WPAN module of NS-2. We used this layer without modification in our implementation.

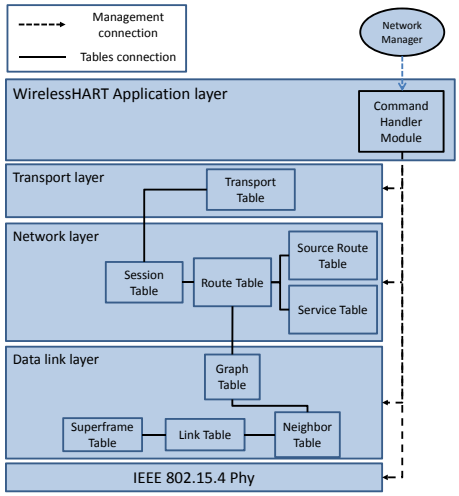


Figure 3.5: WirelessHART protocol stack

3.4.1.2 Data Link layer

We modified the MAC layer of the IEEE 802.15.4 standard (2003 version) module available in NS-2 to support network-wide time synchronization, channel hopping, dedicated slotted unicast communication bandwidth, link layer ACKs and concurrent link activation. Several new MAC layer Management Entity (MLME) primitives, based on the IEEE 802.15.4e (TSCH mode) standard, were also added. The added MLME includes: *mlme_set_slotframe*, *mlme_set_link*, *mlme_set_graph*, *mlme_tsch_mode*, *mlme_listen*, *mlme_advertise*, *mlme_keep_alive*, *mlme_join*, *mlme_activate*, and *mlme_disconnect* [29].

The communication tables shown in the Data Link layer of Figure 3.5 are also implemented. They are manipulated by the NM through the MLME primitives. The tables include:

- ***Superframe table***: This table contains a collection of superframes. Based on the required communication schedule, multiple superframes of different lengths can be configured for each device by filling in this table. The practical superframe length is defined as 2^n s ($-2 \leq n \leq 9$) from 250 ms (2^{-2} s) to 8 min and 32 s (2^9 s) [55].

- ***Link table***: This table contains a collection of links. This table, together with the superframe table, identifies the communication schedule. Based on the traffic rates, multiple links are scheduled for each device in different periods (by specifying the superframe ID to which the link belongs). Each link is specified by the node address, timeslot, channel offset, link type (Normal, Join, Discovery or Broadcast) and link option (Tx-link, Rx-link, or Shared Tx-link).
- ***Graph table***: In a graph table, each graph lists the potential next-hop neighbors that the data can be forwarded to. This table, in collaboration with the route table located in the upper layer, provides sufficient information for routing the packets.
- ***Neighbor table***: Unlike the other communication tables, this table is not filled by the NM. The neighbor table contains the list of neighbors the device can communicate with.

3.4.1.3 Network layer

The Network layer provides routing and secure end-to-end communication for network devices in WirelessHART. To provide secure communication, a Security sublayer is implemented in the Network layer itself. As there is no session layer defined in the WirelessHART stack, a session is defined in the Network layer. To support *Graph Routing* and *Source Routing*, the Route Table and Source Route Table shown in Figure 3.5 are implemented. These tables are manipulated by the NM and are used to deliver a packet to the destination.

1. ***Sessions***: Sessions ensure secure (end-to-end encrypted) communication between two devices in the network (e.g. between the NM and an I/O device or between the Gateway and an I/O device). Four sessions are generally defined in WirelessHART and all the devices (including Gateway and NM) support them [13]. These sessions are the following:
    - A unicast session between the NM and the device. This session is used to manage and configure the network by the NM.
    - A broadcasting session between the NM and all the devices. This session is used to broadcast similar management data to all the devices.
    - A unicast session between the Gateway and the device. This session is used to publish (or subscribe to) the sensor data between the devices and Gateway.

- A broadcasting session between the Gateway and all the devices.

2. *Services*: In WirelessHART, services are used to allocate bandwidth for a specific type of data. The list of services allocated to a field device is stored in a *Service Table* shown in Figure 3.5. In general, four service types are supported by WirelessHART:

   - Maintenance & configuration (default) - This service is used to give the wireless network a minimum overhead bandwidth for basic network control communications [56].
   - Publish - This service is enabled when the device needs to periodically send data or needs to do so on an exception basis. Reporting a sensor reading on a fixed interval constitutes an example of periodic communication [56].
   - Block Transfer - This service is used to send large consecutive blocks of data, such as data log files [56].
   - Event - This service is used to send data packets during unexpected events, such as warnings. These events normally occur infrequently. However, when they do occur, delivery of the data packet is usually urgent. The bandwidth services must therefore be established ahead of time [56].

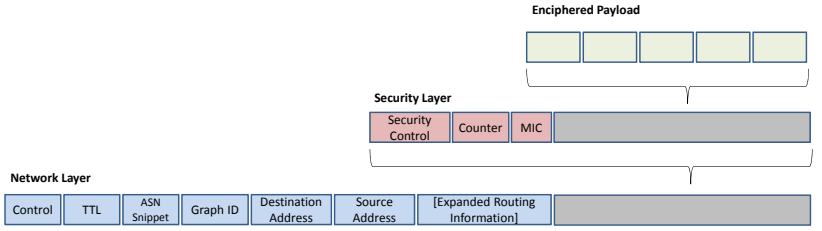


Figure 3.6: WirelessHART Network layer data unit structure

3.4.1.4 Security (sub)layer

WirelessHART provides secure communication between end devices. This is achieved by using cryptographic services in different layers such as the Data Link layer and Network layer. We use the Crypto++ library [57] which supports

various algorithms. The CCM (Counter with CBC-MAC) algorithm with the AES-128 mode of operation is used in our implementation [57].

WirelessHART adopts the CCM* algorithm, which is an IEEE extension to the CCM algorithm. In the IEEE 802.15.4-2006 standard, it is noted that for CCM algorithm, the sum $L + n = 15$ holds, in which L is the placeholder for the size of the message to be enciphered and n is the size of the nonce. As n is 13 in the WirelessHART standard, we get $L = 2$. Since the length of the Message Integrity Code (MIC) is fixed and not equal to 0, there are no constraints for the nonce. Hence the standard is actually using just CCM instead of the CCM* mode.

At the Data Link layer, the integrity of the messages is checked by calculating the MIC in the data link layer, to see if the packets are received from the valid sender. In addition, at the Network Layer, the security sublayer checks the integrity of the messages that travel several hops to the destination by calculating the MIC. The Network Layer Protocol Data Unit (NPDU) is shown in Figure 3.6. The Security Control byte indicates the type of security, which can be *Session Keyed*, *Joined Keyed*, or *Handheld Keyed* for each packet [13].

In order to let the intermediate routers forward the packet to its final destination, the NPDU header is not enciphered. So only the NPDU payload is enciphered to ensure reliable communication. To authenticate the NPDU and to decipher the NPDU payload, a keyed MIC is added to the security sub-layer. The MIC ensures secure communication by checking whether the NPDU received from the correspondent node is forged or not. The CCM* mode is used to generate the MIC, in conjunction with the AES-128 block cipher. At the final destination, the AES-128 engine authenticates the received packet and deciphers the payload.

The authors in [58] analyze the provided security mechanism against well-known threats in the wireless medium, and propose recommendations to mitigate its shortcoming. However, further discussion on security of WirelessHART is beyond the scope of this thesis.

3.4.1.5 Transport layer

The Transport layer ensures that packets are delivered successfully across multiple hops to their final destination. This layer supports either acknowledged or unacknowledged transactions. Unacknowledged service is used for delivering packets that require no end-to-end acknowledgement, e.g. sensor data publishing. On the other hand, the acknowledged service is used to deliver packets that require confirmation of their delivery. The field devices act as slaves during Unicast and Broadcast communications from the NM or Gateway; but they act

as masters (publisher) when sending event notifications to the NM or Gateway, as well as during service request procedures.

For each acknowledged transaction, a new entry is created in the *Transport Table* shown in Figure 3.5. A transport pipe that connects two devices is constructed across the network. Each WirelessHART device might track multiple transport pipes. The Gateway and the NM often track many transport sessions with each field device. For example, when it uses the acknowledged broadcast initiated by the Gateway or NM, the Transport Layer tracks the reception of acknowledgment from all the affected devices. The Transport layer also supports the aggregation of multiple HART commands in a single transaction. This method is especially useful when sending (or reading) several configuration commands to (or from) a network device.

3.4.1.6 Application layer

The Application layer of WirelessHART is a command based layer. Commands, the basis of HART communications, are sent from gateway or field devices. Each command can be identified by a command number, which determines the content of the message. The WirelessHART commands are a collection of commands in the range 768-1023, which can be used to support network management and gateway functions [13]. The commands implemented can be classified into the following categories: managing superframes and links commands, managing graph and source routes commands, bandwidth management commands, network health reporting and status commands.

3.4.2 WirelessHART network management algorithm

The WirelessHART NM uses centralized network management techniques for communication scheduling and managing routes. However, it does not define any specific algorithm for the NM. The management algorithm introduced in [55] is one of the few network management algorithms that address both routing and communication scheduling. We choose this algorithm for our implementation. According to [55], each time a new node joins the network, the algorithm is executed and it tries to find new *Uplink*, *Broadcast*, and *Downlink* graphs, and defines communication schedules for the new device. This process is done incrementally, until all the nodes join the network.

This section considers the implementation of the network management algorithms, by discussing their four most important parts: the joining procedure,

graph and route definition, communication scheduling, and finally, the service request procedure.

3.4.2.1 Joining procedure

The joining sequence of a new device is shown in Figure 3.7. Nodes that have already joined the network periodically send advertisements used for synchronization purposes and to inform nodes that want to start the binding process about the superframes' structure. Nodes that want to participate in the network must know the (time)position of the join timeslots in the superframe; in these join timeslots nodes are allowed to send join requests. The new device that intends to join the network listens consecutively on all physical channels for a while. It selects the best advertiser/candidate based on certain predefined criteria and sends the join request to the selected advertiser. The join request contains *Report Neighbour Signal Levels* (command 787) as well as other information. The new device includes the advertiser Graph ID in the network header. The join request is forwarded toward the Gateway/NM. The NM, upon receiving the join request, allocates network resources (such as graphs and links) based on the management algorithm and sends a join response/activation command to the new device, after all necessary network resources are configured and reserved along the path. The NM then sends the join response including three commands, *Write Network Key* (command 961), *Write Device Nickname Address* (command 962), and *Write Session* (command 963). Finally, the NM sends the commands to write the superframe and links in the communication table of the new device. These are the only commands, besides the join response, that can be proxy routed.

3.4.2.2 Graph and routes definition in the network

To address different communication requirements, three types of routing graphs are defined in any WirelessHART network.

- Uplink graph is a graph connecting all devices to the gateway. It is used to forward both the devices' management data and process data to the gateway.
- Broadcast graph connects the gateway to all devices. It can be used to broadcast either common data or control data to the entire network.
- Downlink graph is defined per device. It is used to forward unicast messages from the gateway to each individual device.

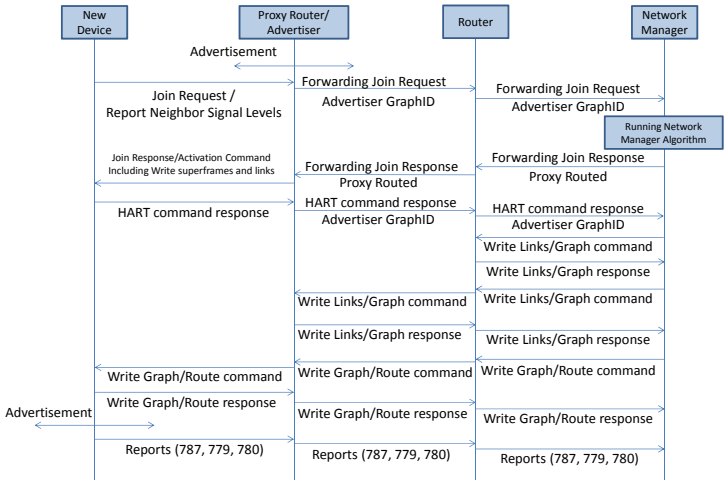


Figure 3.7: Joining Process

To construct these graphs in a reliable manner, the algorithms *"Constructing Reliable Broadcast Graph"*, *"Constructing Reliable Uplink Graph"* and *"Constructing Reliable Downlink Graphs"* in [55] are implemented in the NM. These algorithms are designed to maintain the maximum number of reliable nodes in the graphs while achieving good network latency.

3.4.2.3 Communication scheduling and channel management

After constructing the Uplink, Broadcast, and Downlink graphs, the algorithms *"Constructing Data Communication Schedule"* and *"ScheduleLinks"* in [55] are used to construct the data communication schedules and to define links and superframes. These algorithms are implemented in the NM. These algorithms use the Fastest Sample Rate First policy (FSRF) to schedule the devices' periodic publishing and control data. The construction is based on the reliable graphs. In Figure 3.8, a sample connection is shown in which the NM has allocated the resources from the sensor node (37) to the actuator node (45). The sensors publish process data using Commands 1, 3, 9, etc. Command 79 is used to write data to the actuators [59]. In this work, similar to what is described in [59], we assume that WirelessHART supports *Control in the Host* or *Control in the Gateway*.

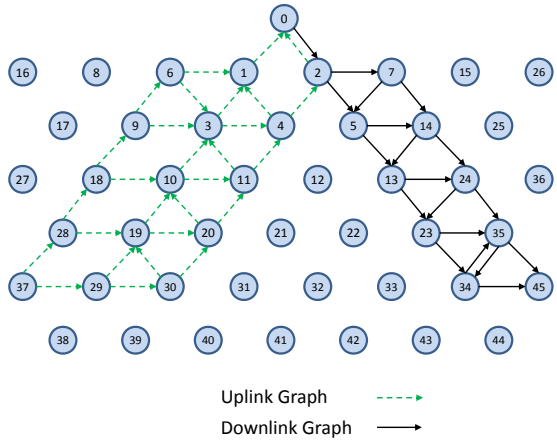


Figure 3.8: A sample connection establishment between nodes 37 and 45

3.4.2.4 Service request procedure

A device that needs to establish a connection with the other devices, e.g. actuators, sends out *service request* (command 799) to the NM asking for additional bandwidth. The service request handling procedure is illustrated in Figure 3.9. The NM allocates sufficient bandwidth along the uplink graph from the sensor to the gateway and along the downlink graph from the gateway to the actuator, by adding links in a new route or an existing route. This process may take some time. Upon completion, the NM replies to the requesting device.

3.5 WirelessHART Validation

To validate the WirelessHART simulator implementation, we need a real WirelessHART network to generate the traffic patterns. A testbed has been purposely designed in order to emulate a typical industrial environment, i.e. an instrumented steam generation process at the Idrolab of ENEL in Italy. A similar network has been set up in the NS-2 simulator. The network topologies of real and simulated setups are shown in Figure 3.10, in (a) and (b) respectively. The collected traffic from the real network is used to (i) validate the correctness of the implementation of the WirelessHART stack and (ii) to confirm that the NM

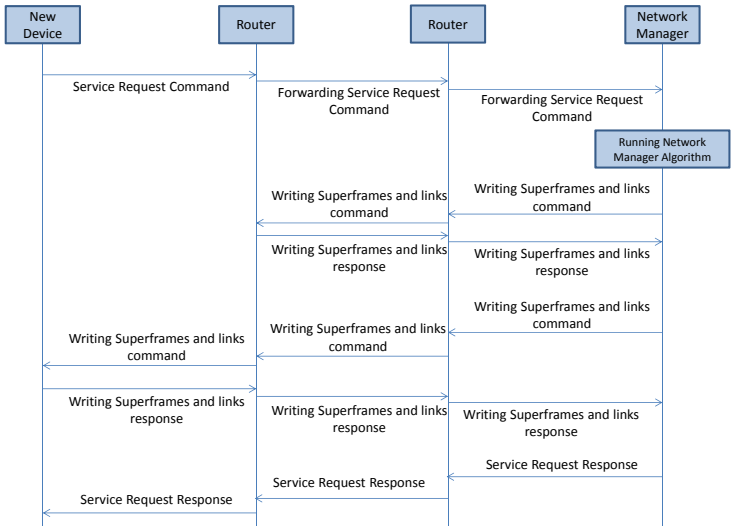


Figure 3.9: Service request process

used in the simulator manages the network in a similar fashion as the real NM used in industry.

3.5.1 Real world experimental setup

The Idrolab test plant [60] is depicted in Figure 3.11. The instruments are not actually attached to the plant, but they flank the legacy of existing wired control systems in order to experience similar *harsh* environmental conditions. This WirelessHART network comprises:

- A PC-based Host station implementing a Mobus/TCP and an OPC client, both of them purposely implemented in LabVIEW. The Modbus/TCP server is embedded in the Wireless HART Gateway while the OPC server is implemented by means of the HART server, which translates OPC messages into HART/IP requests and responses. In addition, the PC can directly inject HART/IP traffic in the network.
- A WirelessHART Gateway with GW-NM-SM functionalities from Pepperl+Fuchs (WHA-GW) based on the Dust Networks' SmartMesh IA-510

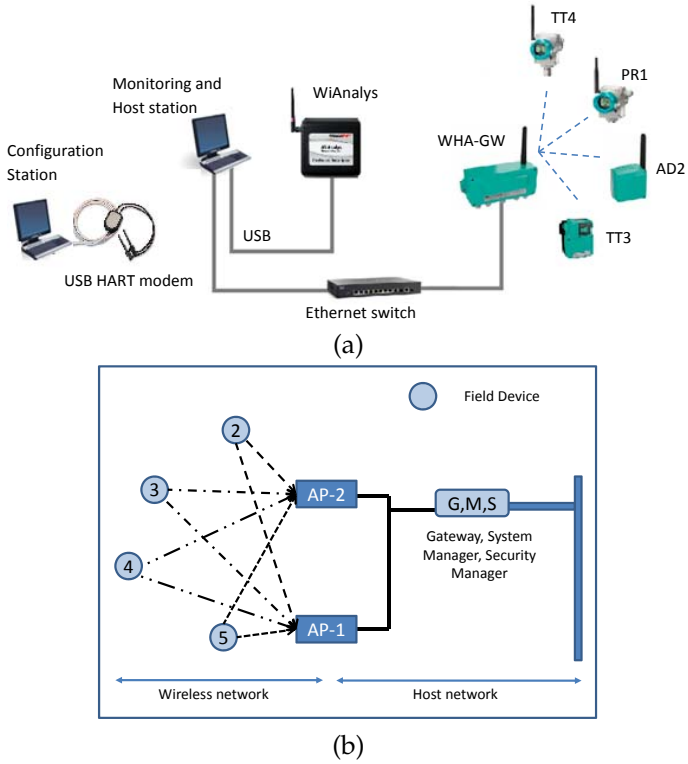


Figure 3.10: Network topology for (a) real and (b) simulated setups

device. It provides an Ethernet connection towards the host application, supports HART/IP and Modbus/TCP protocols and handles ModbusRTU (not used in this work).

- A pressure transmitter from Siemens (Sitrans P280, PR1).
- A WirelessHART adapter from Pepperl+Fuchs (WHA-ADP, AS2), which can acquire the signal coming from a legacy 4-20 mA transmitter.
- A temperature transmitter from P+F (WHA-UT, TT3); the actual sensing element is an external PT100.

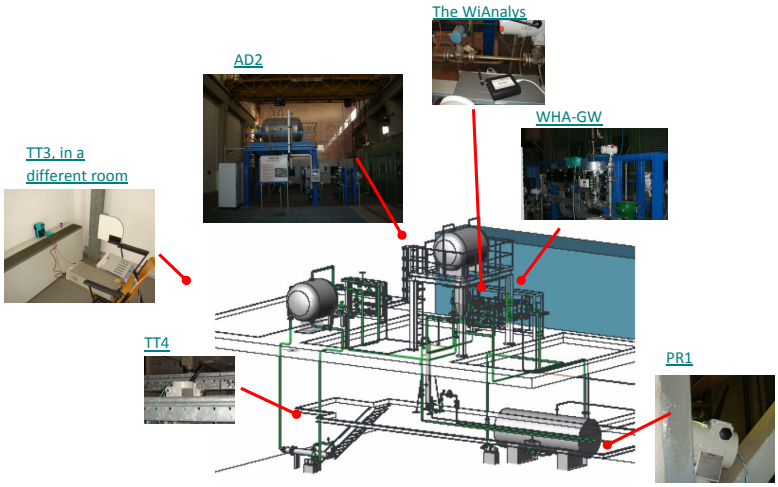


Figure 3.11: Idrolab test plant

- A temperature transmitter from Siemens (Sitrans TF280, TT4); the actual sensing element is an external PT100.
- A PC-based Monitoring station tool to collect data exchanges over the Ethernet link and over the air, implemented by the Host station PC.
- A PC-based Configuration station used to commission the network leveraging on a USB HART modem from Microflx for field devices' NetworkID, JoinKey and operating parameters.

We need to collect the traffic patterns from the real network to be able to validate the simulator. We use the open source tool Wireshark to analyze the Ethernet traffic. Regarding the over-the-air traffic, two possible approaches can be practically adopted based on the channel hopping mechanism. In the first one, the *blacklisting* feature offered by the WirelessHART protocol can be exploited to limit the number of radio frequency channels used for hopping without losing generality. For instance, one can use this feature to limit the radio channels to a subset and use a reduced number of low-cost IEEE 802.15.4 compliant protocol analyzers. For example, we exploited three low-cost USB connected radio probes, (UZBee devices from Flexipanel[61]) to collect traffic

logs from three active channels (formally 22, 23 and 24). These three traffic logs were then merged, based on the collected messages' timestamps.

The second approach exploits fifteen transceivers (WirelessHART only supports fifteen channels) to simultaneously scan all the ISM band at 2.4 GHz. For instance, we also use the the WiAnalys tool, developed by the HCF consortium [62], that hosts an FPGA for managing the IEEE 802.15.4 transceivers. In both cases, a post-process software running on the monitoring station decodes raw message logs and recognizes different stack levels. The results presented in this chapter refer in particular to the WiAnalys tool.

3.5.1.1 Addressing security aspects

During the data collection from the real network, there is a need to address the security authentication in the Data Link layer and to decrypt the NPDU. We manage to authenticate the messages with the MIC. To calculate the MIC of the DLPDU during joining process, we use the well-known public key 7777 772E 6861 7274 636F 6D6D 2E6F 7267 hexadecimal, which is the ASCII value sequence of the 16 character string of the HART Foundation's web address: www.hartcomm.org. We decrypt the NPDU from the Join Request message using the Join Key of each device, which is known in advance. After the successful decryption of the Join Requests, we follow the Initialization command, which contains the new Session keys that will replace the Join Key. We also follow the new Network key, which will replace a well-known key for calculating the MIC at the Data Link layer for each I/O device. Each following message is then first decrypted and checked if it contains the command for changing the Session keys or the Network Key, in which case we save the new keys for that particular node and start using them with the next message.

3.5.2 Simulation model and parameters

In the NS-2 simulator, we set up a similar network with four field devices, which are connected to the Gateway through two Access Points (APs), as shown in Figure 3.12. It is a snapshot taken from nam, the Tcl/Tk-based animation tool for viewing network simulation traces in NS-2. We assume that the connection between APs and the Gateway is wireless. The details of the simulation parameters are presented in Table 3.1 and Table 3.2. We choose the shadowing radio propagation model as it is a more general model allowing for more realistic predictions with multi-path and fading effects [63]. The shadowing model consists of two parts as shown in Equation 3.2. The first part is the path loss model that

predicts the mean received power at distance d and d_0 as a reference-distance, while the second part reflects the variation of the received power, which is a Gaussian random variable with zero mean and standard deviation σ_{dB}. σ_{dB} is referred to as a shadowing deviation and its value for two different environment are provided in Table 3.1.

$$\left[\frac{P_r(d)}{P_r(d_0)}\right]_{dB} = -10\beta \log\left(\frac{d}{d_0}\right) + X_{dB} \tag{3.2}$$

The simulation scenarios are implemented in NS-2, by using Tcl scripts. The scripts comprise commands and parameters for simulator initialization, node creation and configuration, such as *startWHGateway*, *startWHAccessPoint*, *startWHDevice*, or *requestService* commands. The commands can be used respectively to start a Gateway/NM, Access points, Field devices, and to request more bandwidth to communicate with the other devices.

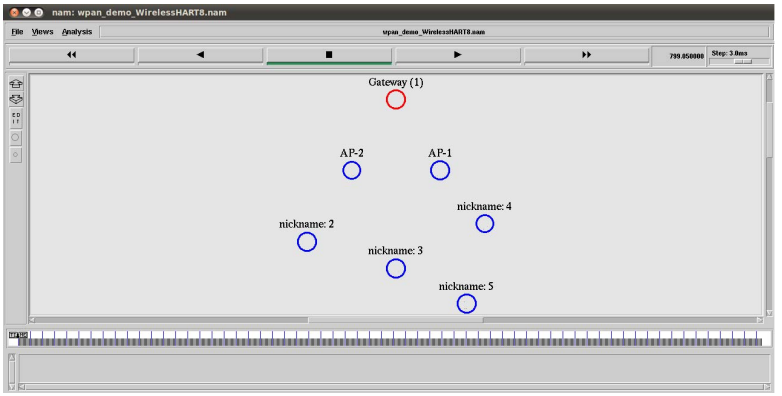


Figure 3.12: The network topology from animation tool of NS-2 simulator (nam).

3.5.3 Validating the WirelessHART stack

To validate the WirelessHART stack implemented in the simulator, the simulated NM is replaced by the real NM. The radio frequency and the time stamp at which the sniffer receives the packet, identify each packet in the traffic log file. All traffic generated by the real NM is filtered from the traffic log file. This

Table 3.1: NS-2 simulation parameters

Parameter	Value
Number of Routers	Gateway, two access points
Number of I/O devices	4
Simulation area	100×100
Minimum Superframe size (Real network)	128 slot
Minimum Superframe size (Simulated network)	200 slot
Data rate	250 kb/s
Frequency Band and channel	2.4 GHz, 11–26 channels
Radio range	$\approx$ 40 meters
Radio propagation model	Shadowing model
Path loss exponent	2.0
Shadowing deviation (dBm)	5.7(Engineering building) & 8 (corridor)
Reference distance	1.0 m
Mac retransmission	3
Application traffic model	CBR

traffic includes the joining response, activation commands and all management commands that manipulate the communication table as well as different tables in the field devices. For this traffic, a virtual NM generates corresponding events in the simulator at the same frequency and time. Thus the times of the simulator and the real network get aligned. The neighboring field devices of the virtual NM receive the packets and forward them to the destination node. At the destination node, the packet traverses through each layer of the simulated stack and reaches the command handler in application layer. By checking the validity of the commands received, it is possible to verify the implemented WirelessHART stack.

3.5.4 Validating the WirelessHART Network Manager

In order to validate the simulated WirelessHART NM, we need to show that the implemented NM manages the network similar to the real NM. To this end, we create a network with the same number of field devices in the simulator as

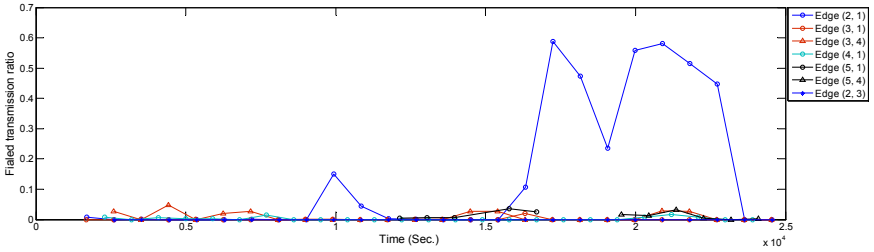
Table 3.2: Periodic messages rates

Item	Parameter	Value	Transmission type
Simulated periodic management data	Neighbor Health List	30 s	Acknowledged unicast
	Neighbor Signal Level reports	30 s	Acknowledged unicast
	Advertisement rate	4 s	Un-Acknowledged broadcast
Real network periodic reports and advertisement	Advertisement rate	1.28 s	Un-Acknowledged broadcast
	Device health report	914 s	Acknowledged unicast
	Neighbor Health List report	914 s	Acknowledged unicast
	Neighbor Signal Level report	914 s	Acknowledged unicast
Application Data for Real and simulated network	Sensor Data rate	4 s & 60 s	Acknowledged unicast

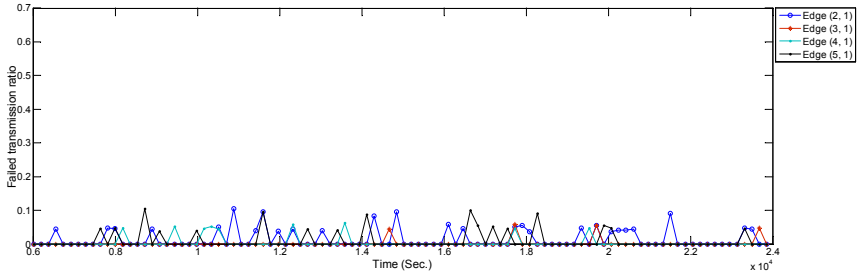
they are in the real test-bed scenario. By measuring the management overheads, reliability, end-to-end delay and communication scheduling of both the simulated network and the real network and by comparing the collected statistics, we show that the two NMs function almost in a similar manner and thereby we can validate the simulated WirelessHART NM. The details are described in Section 3.6

3.6 Experimental analysis of real and simulated networks

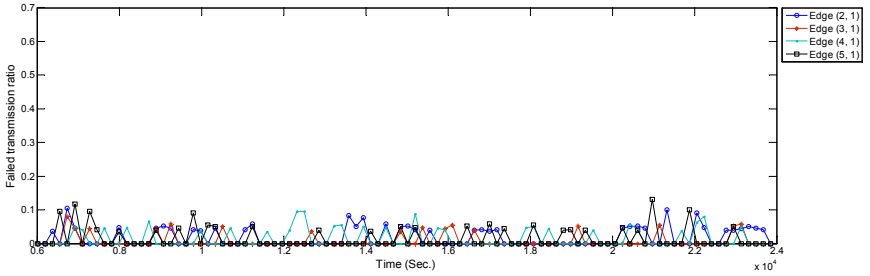
We collect traffic patterns from the real network installed at the testbed in the Idrolab for about 24000 seconds. Initially, all field devices are placed within 40 meters from the Gateway and they form a star network with the Gateway. After some time, node 5 is moved away from the Gateway so that a 2 hop network is formed. In the simulator, we consider a similar placement, but with a fixed



(a) Real network



(b) Simulated network with shadowing deviation of 5.7 dB



(c) Simulated network with shadowing deviation of 8 dB

Figure 3.13: The failed transmission ratio on different edges over time

position for node 5, after which a star network is formed. Since node 5 is located far away from the access points, as shown in Figure 3.12, node 5 sometimes uses node 4 as an intermediate node and a 2 hop network is also formed.

3.6.1 Reliability in the network

In this section, we evaluate the behavior of the real network and the simulated network in terms of reliability. We use the *Neighbor Health List* report to evaluate the quality of connections between the network field devices. These reports provide the statistics for linked neighbors. Figure 3.13 (a) shows the percentage of failed transmissions on different edges in the real network over time. A very small percentage of transmissions fails, except for the Edge (2,1) between node 2 and the Gateway. When the connection quality drops between node 2 and the Gateway, the NM defines more links between node 2 and node 3. Some of the traffic of node 2 to the Gateway is forwarded through node 3. As a result, the problem is fixed. Figure 3.13 (b) and (c) show the percentage of failed transmissions on different edges in the simulated network over time, with shadowing deviations of 8 dB and 5.7 dB that correspond to corridor and

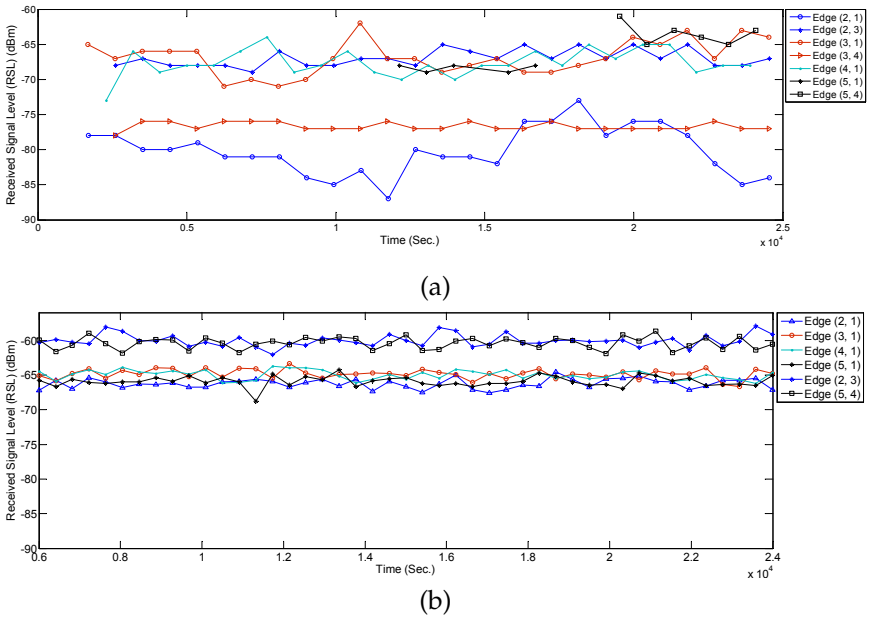


Figure 3.14: The average of Receive Signal Level (RSL) on different edges in the network over time (a) for real network and (b) for simulated network

engineering building [64] environments respectively. As the deviation increases in the shadowing model, the packet drop increases likewise.

Figure 3.14 (a) and (b) display the average of Receive Signal Levels (RSL) on different edges in the real and simulated network over time. RSLs considerably differ from one another in real networks, whereas in simulations they are quite close. We also see in Figure 3.14 (a) that the RSL between node 2 and the Gateway varies a lot over time. This variation also justifies the earlier mentioned statement that the NM defines more links between node 2 and node 3 to overcome the problem in the connection between node 2 and the Gateway. For the connections between other nodes and the Gateway, the RSL values in the simulation are very close to the real values. They deviate between -65 dBm and -70 dBm.

3.6.2 Communication schedules and network throughput

Figure 3.15 and Figure 3.16 show the global matrix of the reserved communications by the NM in the real and simulated network scenarios. The real

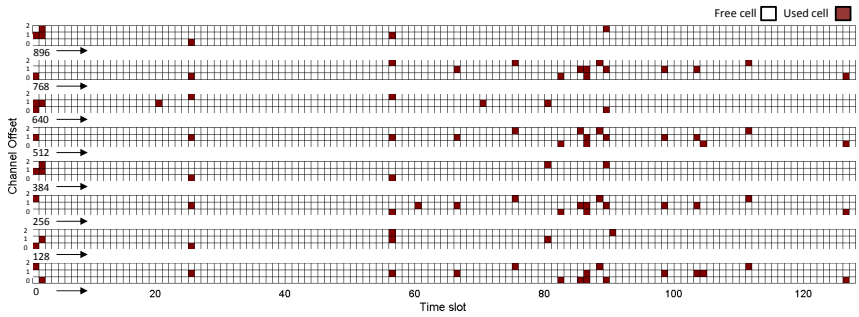


Figure 3.15: The global matrix of the current slot/channel usage for the real WirelessHART network (the combination of superframes with size 128, 256 and 1024 timeslots)

network has a combination of superframes with size 128, 256 and 1024 timeslots, whereas the simulated network has a superframe length of 200 timeslots. The NM schedules interference-free cells to transmit management traffic or sensor data. We can see that the allocation patterns are quite different. This is because the simulated NM constructs the communication schedules based on the proposed algorithm in [55], where it allocates from source to destination

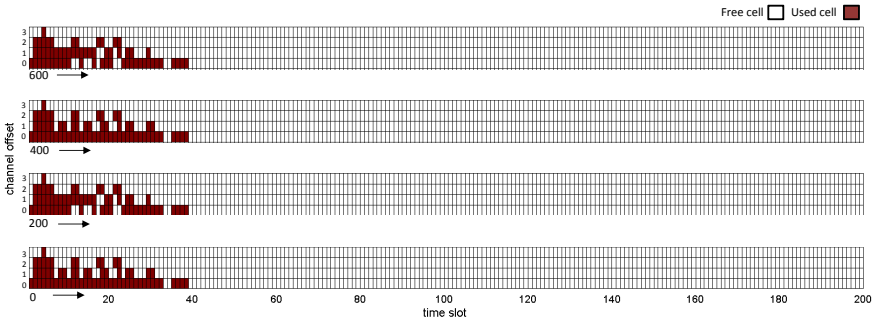
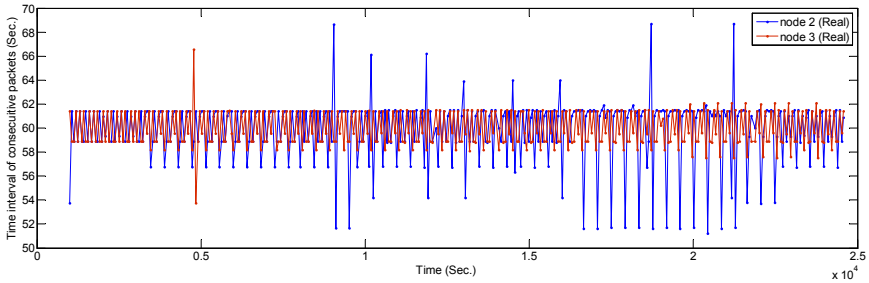


Figure 3.16: The global matrix of the current slot/channel usage for the simulated WirelessHART network

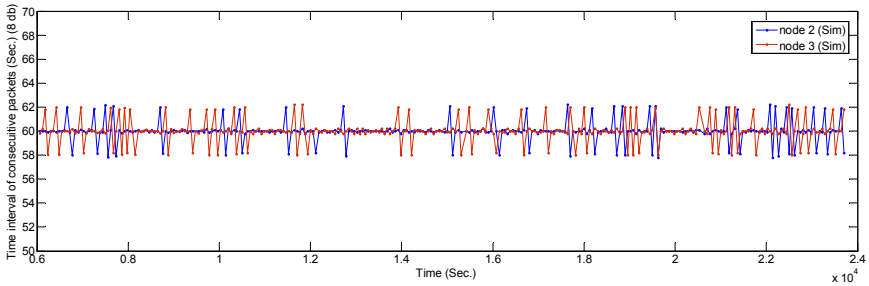
each link on the paths in a depth-first manner. Hence it allocates the earliest available timeslot to each link and updates the schedule matrix as well as each effected node's schedule accordingly. In the real NM, the undisclosed algorithm seems to allocate cells randomly. Since the simulated NM allocates more links between devices, the communication schedule in Figure 3.16 is denser than the communication schedule in Figure 3.15. Allocation of more cells might affect management efficiency due to (i) joining process delay and overhead, (ii) bandwidth allocation based on service requests, (iii) coping with node/edge failure in the network. It might also affect power consumption and end-to-end latency. In such a case, allocating more cells (over provisioning) will increase the energy consumption of the nodes. On the other hand, it will improve end-to-end latency.

3.6.3 Real-time guarantee

To evaluate the end-to-end data delivery delay, we measure at the Gateway the time interval between the consecutive received packets, which are sent by the field devices during the network operating time. Figure 3.17 (a) and (b) display the results for nodes 2 and 3 with a constant publishing period of 60 seconds in the real and simulated network respectively while Figure 3.18 (a) and (b) show the results for nodes 4 and 5 with a period of 4 seconds. The simulated sensor nodes publish the data at the specified rates following the Constant Bit Rate (CBR) traffic model employed in NS-2. The required resources to support



(a) Real network



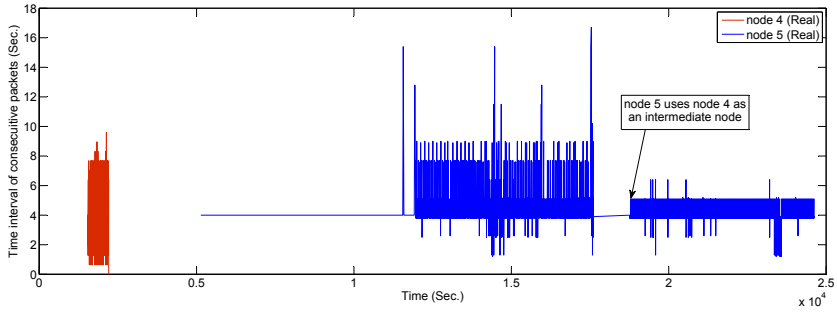
(b) Simulated network with a shadowing deviation of 8 dB

Figure 3.17: Time interval of the consecutive received packets for node 2 and 3

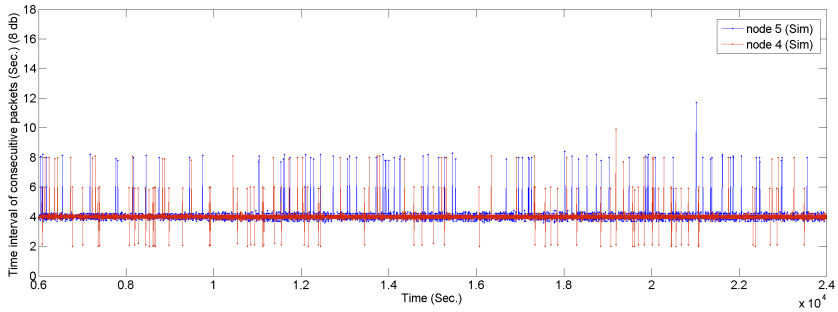
these traffic characteristics are reserved beforehand, along the path between the sensors and the Gateway, for both real and simulated networks.

The results show that in the simulator, the real-time communication requirements are addressed much better than in the real network. The presence of external interference in harsh industrial environments could explain this difference. This causes more packet drop and possibly more retry at the MAC layer in the real network.

In Figure 3.17 (a), we see that the connection quality (timeliness) between node 2 and the Gateway drops after some time while the end-to-end delay increases. Then the NM, at around 2.1×10^4 s, defines more transmission links between node 2 and node 3 and some of the traffic of node 2 toward the Gateway is forwarded through node 3, bringing down the end-to-end delay. In addition, we see in Figure 3.18 (a) that the connection quality between node 5 and the Gateway drops after a certain time. This is caused by an intentional increase



(a) Real network



(b) Simulated network with a shadowing deviation of 8dB

Figure 3.18: Time interval of the consecutive received packets for node 4 and 5

of the distance between these nodes in the real network. At around 1.8×10^4 s, the NM considers node 4 as an intermediate node between node 5 and the Gateway and writes several links between node 4 and node 5. Afterwards, the end-to-end delay is reduced significantly. In the simulation, we did not move node 5 and so no such variations are seen.

In an industrial environment, we expect large shadowing due to the presence of heavy machinery, which typically causes a positive biased shadowing effect. The shadowing effect can vary according to different industrial setups. In the simulator, we choose the shadowing model. In order to simulate a harsh industrial environment, we need to propose a channel model that represents that environment, more accurately.

3.6.4 Energy Consumption in the Network

In this section, we evaluate the energy consumption of network nodes in real and simulated scenarios. We measure the total consumed energy at every node during the 24000 s time period of the network operation. The periodic management messages generated by each device in the WirelessHART network consist of network health reporting and status commands (i.e., WirelessHART command 779, 780, and 787) and advertisements. Management and application data messages for WirelessHART are listed in Table 3.2.

Table 3.3: Energy-consumption parameters

Parameter	Value	Parameter	Value
Radio chip	Dust [65]	Supply Voltage	3.76 V
Transmit power (0 dBm)	20.303 mW	Receive power	16.92 mW
Listen power	16.92 mW	Receive a packet	4.5 mA
Transmit at 0dBm	5.4 mA	TsRxWait	2.2 ms
TsAck (26 bytes)	0.832 ms	TsCCA	0.128 ms
TsRxTx (TxRx turnaround)	0.192 ms	TsMaxPacket (133 bytes)	4.256 ms

Table 3.4: Energy-consumption per transaction

Notation	Formula	Value
Acknowledged Tx	TsCCA*Listen power + TsMaxPacket*Transmit power + TsAck* Receive power	102.6 μJ
Acknowledged Rx	TsMaxPacket*Receive power + TsAck*Transmit power	88.90 μJ
Broadcast Tx	TsCCA*Listen power + TsMaxPacket*Transmit power	88.57 μJ
Broadcast Rx	TsMaxPacket*Receive power	72.01 μJ
Idle	Rx TsRxWait*Listen power	37.22 μJ

The specific values of the parameters used in the calculations are listed in

Table 3.3. The timing parameters are illustrated in Figure 3.2. Table 6.4 shows the energy consumption required for each type of transaction[1]. In addition, the idle listening energy at an unused scheduled link is calculated: the energy consumed by the receiver while waiting for a message.

Table 3.5 also lists the energy consumed by each node as well as by the Gateway in both the real and the simulated network. In the simulated network, the energy consumed by the nodes is more than in the real network. Part of this difference can be explained by the fact that in the simulator, we defined more links between the nodes. Furthermore, the considered management message rate is different in the simulator. We also see that the energy consumed by node 4 is higher than the energy consumed by the other nodes in the simulator. This is because node 4 is considered an intermediate node in the uplink and downlink graph for node 5, as it is located far away from the access points.

Table 3.5: Energy-consumption in the network (in 25,000 s) during normal operation.

Scenario	Item	GW-NM	node 2	node 3	node 4	node 5
Real (Dust)	Total Energy (without idle listening)	9.80 J	1.93 J	1.98 J	1.96 J	0.64 J
	Total Energy (without Adv and idle)	0.58 J	0.22 J	0.27 J	0.34 J	0.18 J
Simulation (Dust)	Total Energy (without idle listening)	7.60 J	1.56 J	2.74 J	2.85 J	1.64 J
	Total Energy (without Adv and idle)	1.47 J	0.19 J	0.72 J	1.15 J	0.60 J

3.6.5 Evaluating Management Efficiency

In this section, we evaluate the I/O device joining procedure as well as the service request procedure by measuring the delay and communication overhead in both the real and simulated WirelessHART networks.

3.6.5.1 Performance during node joining

In WirelessHART, as discussed in Section 3.4.2.1, the joining process includes scanning the channels for a while for router discovery, sending the join request

[1] in this calculation we assumed the energy consumption in Tx/Rx turnaround, and the processing energy can be neglected

to the routers and receiving the management communication resources and related graphs/route information. As shown in Figure 3.7, the joining process is considered to start from the moment that the node sends the join request till the moment that it begins to broadcast the advertisements and send the periodic reports toward the NM. However, in our comparison of the joining process in simulated and real networks, we consider the total delay and overhead of the management resources reservation without accounting for scanning delay.

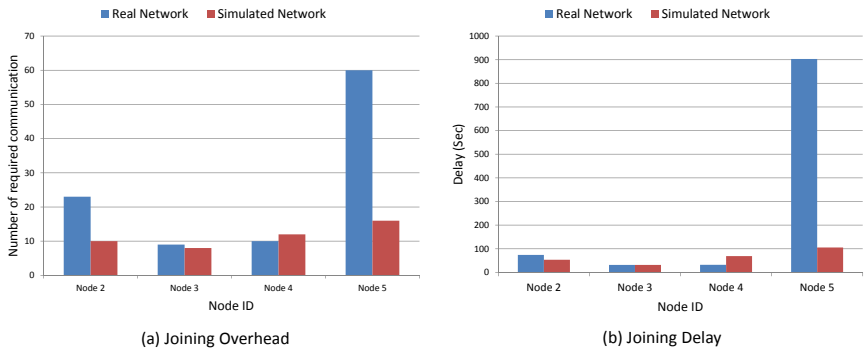


Figure 3.19: Field device joining overhead (a) and delay (b) (real vs. simulated WirelessHART network).

Figure 3.19 (a) and (b) display the delay in, and the number of communications required (number of messages sent) for, I/O device joining. There is no considerable deviation in delay and overhead in both scenarios except for node 5, whose position has been changed in the real experiments.

3.6.5.2 Service request procedure between I/O devices and Gateway

In this evaluation, we compare the management efficiency of service request procedures by measuring the delay and the number of communications required for reserving communication resources between field devices and the Gateway. Figure 3.20 shows that for nodes 2 and 3, the NM does not allocate any communication resources in the real scenario, as it defines sufficient resources during the network setup. The overhead of all nodes in the simulator exceeds the one in the real scenario. This is because the NM assigns more links in the simulator (Section 3.4.2). Hence, more messages are sent in the simulator than

in the real network. This also makes the delay in simulations much lower than in the real scenario, except in the case of node 5, as links are already assigned. Node 5 uses node 4 as the intermediate node in the simulation. In the real scenario, node 5 initially communicates directly with the Gateway, but after it has been moved further away from the Gateway, the NM considers node 4 as an intermediate node for node 5 and allocates new resources between node 4 and 5. This increases the overall delay.

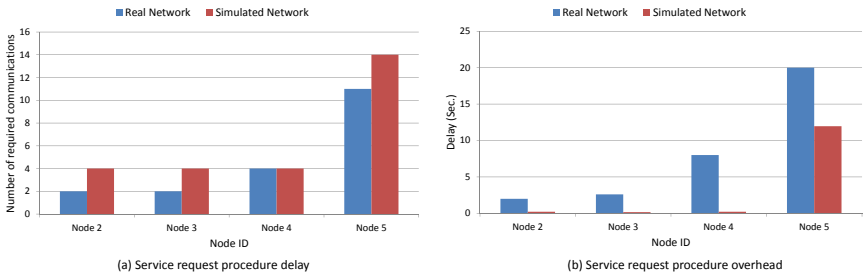


Figure 3.20: Service request procedure (a)overhead and (b) delay (real vs. simulated WirelessHART network).

3.6.6 Summary

We found that the network management algorithm greatly affects the performance of the WirelessHART network, namely during node joining, the service request procedure, data delivery latency, and when coping with node/link failure. Consequently, when applying other system management algorithms results may differ.

3.7 Experimental analysis of a multi-hop mesh network in simulator

In this section, we show some results from a multi-hop mesh network that is used to demonstrate the usability of the simulator. In these experiments, we consider a simulation area of a size of 150 m $\times$ 150 m, with field devices placed away from each other at a distance of 10 m, as shown in Figure 3.21. The

transmission range is set to approximately 15 m. We use the two-ray ground model as a radio propagation model [63]. The network consists of one gateway, two access points, and 43 field devices. All the results reflect the average values achieved after the experiments were repeated several times.

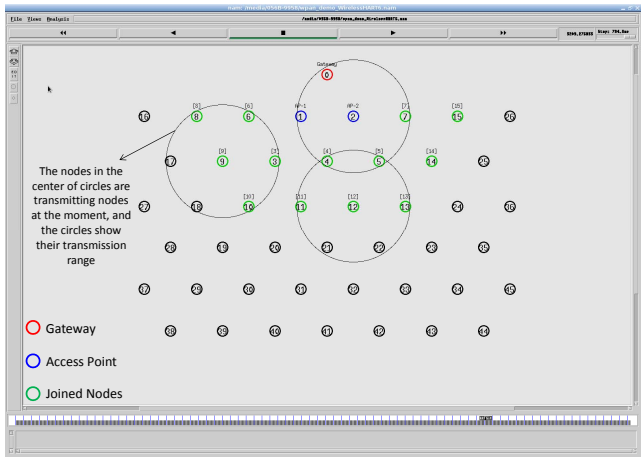


Figure 3.21: A sample multi-hop mesh network topology in NS-2 simulator

Node and link failure in the network: In this part, we demonstrate the behavior of the system in case of link and node failure. Figure 6 shows a sample downlink graph toward node 45. Figure 3.22 shows the failure of node 24 in the network, as well as how the system manager copes with this node failure by defining new links and by deleting unnecessary links.

Figure 3.23 (a) and (b) show how the system behaves while link failures are being varied. The link failures are introduced randomly on different hop levels. We increased the number of link failures from 1 to 10 and measured the delay in, and the number of required communications for, coping with the link failures. Even though the network may still work when the graphs are unreliable, the implemented management algorithm tries to establish a reliable graph and to construct a new schedule. This causes a relatively high delay and a large number of communications.

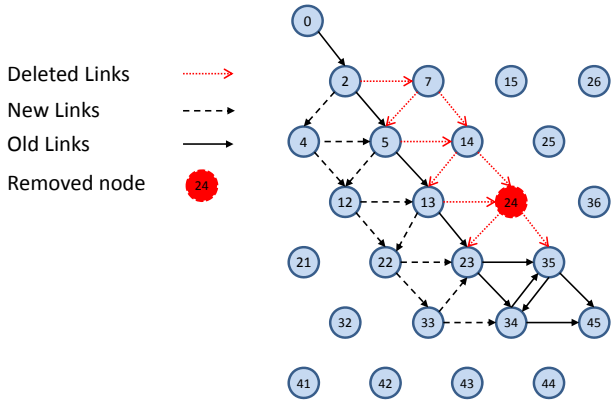
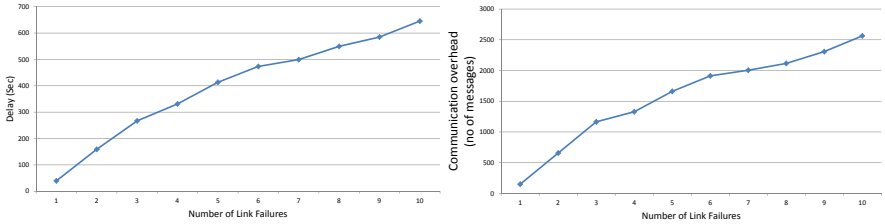


Figure 3.22: Node failure sample



(a) (b)

Figure 3.23: Network maintenance (a) delays and (b) overhead

3.8 Usage of WirelessHART implementation

In this section, we provide some examples of usages of the WirelessHART simulator that can help network/control engineers as well as protocol designers.

3.8.1 Feasibility study of WirelessHART in different application scenarios

Network engineers can check the feasibility of using the WirelessHART network in various application scenarios with different requirements very easily with our

implementation. By having predefined application scenarios, i.e. the number and position of nodes as well as the sampling rates of sensor nodes, they can study whether the NM can allocate sufficient resources/bandwidth in those scenarios or not. The network designers can also study the network coverage and connectivity characteristics. For example, by considering the network topology in the NM, additional routers might be deployed in the network, if the sensor and actuators are not covered or if the constructed graphs are unreliable. The control engineers can also study the possible data delivery delay in the control loop.

By using the Tcl scripts, such as *WHnode-down*, *WHlink-down*, *WHnode-up*, or *WHlink-up*, which the WirelessHART simulator provides, network designers can simulate node/link failure and introduce dynamicity in the network. They can evaluate whether the WirelessHART protocol and network management algorithm considered can cope with the dynamicity and disturbance in the network or not.

3.8.2 Evaluating other wireless protocols or WirelessHART itself

Protocol designers can use this implementation as a tool to compare the performance of other wireless protocols with WirelessHART. They can easily extend this simulator to develop new protocols or simulate other existing protocols like the ISA100.11a standard. That way, they can assess whether a distributed management approach is better in coping with highly dynamic situations in a timely manner than centralized approaches like WirelessHART, as we did in [19].

Designers can also test other network management algorithms and compare their performances. One can also modify the WirelessHART stack implementation and test various mechanisms used in different parts of the stack. For instance, the blind channel hopping and global blacklisting techniques, used in WirelessHART to mitigate external interferences and multi-path fading, can be substituted with other techniques, e.g. local blacklisting, and the performance of those schemes can be measured.

3.9 Conclusion and future works

We presented the complete implementation of the WirelessHART standard in the NS-2 network simulator and showed how this implementation can be used

as a reference point to evaluate other wireless protocols, as well as to improve the WirelessHART stack and network management algorithm. Using sniffed traffic from the real WirelessHART network, we validated (i) the WirelessHART protocol stack and (ii) the Network Manager. Validation of the Wireless protocol stack by using the captured traffic from other real WirelessHART networks is feasible since our implementation effort follows the standard. However, since the standard does not specify the specific management algorithm, different implementation efforts have different characteristics. By comparing the real and simulated network managers, we found that differing network management algorithms might affect the performance of the WirelessHART network. However, it is expected and observed that the main requirements of the WirelessHART protocol, namely the provision of reliable and also real-time communication, should be fulfilled by different network manager algorithms. Consequently, when applying other network management algorithms, results may differ. Empirical analysis showed that the simulated results are quite close to the results obtained from real networks. Hence we can make very realistic simulations with our implementation. We also demonstrate the versatility and usability of our implementation by showing some further evaluation results in diverse scenarios.

CHAPTER 4

D-MSR: A Distributed Network Management Scheme for Real-time Industrial Wireless Automation

Current wireless technologies for industrial applications, such as WirelessHART and ISA100.11a, use a centralized management approach where a central network manager handles the requirements of the static network. However, such a centralized approach has several drawbacks. For example, it cannot cope with dynamicity/disturbance in large-scale networks in a real-time manner and it incurs a high communication overhead and latency for exchanging management traffic. In this chapter, we therefore propose a distributed network management scheme, D-MSR. It enables the network devices to join the network, schedule their communications, establish end-to-end connections by reserving the communication resources for addressing real-time requirements, and cope with network dynamicity (e.g., node/edge failures) in a distributed manner. According to our knowledge, this is the first distributed management scheme based on IEEE 802.15.4e standard, which guides the nodes in different phases from joining until publishing their sensor data in the network. We demonstrate via simulation that D-MSR can address real-time and reliable communication as well as the high throughput requirements of industrial automation wireless networks, while also achieving higher efficiency in network management than WirelessHART, in terms of delay and overhead.

4.1 Introduction

Certain Quality of Service (QoS) mechanisms are used by communication networks to meet the real-time requirements. These mechanisms can generally be categorized into: (i) traffic classification and (ii) resource reservation. The traffic classification mechanism can be used for channel access and packet delivery along the path between the endpoints, by labeling the packets with a priority value and placing them on the corresponding queue in the path. The resource reservation technique allocates the communication resources along the path between two end-points for a specific traffic or class of traffic to achieve the desired QoS requirement [16].

In addition to real-time communication, reliability is also an essential requirement for communication in harsh industrial environments in the presence of interference. The links quality between a source and destination node can heavily influence the success of the delivery of sensor data to the destination when the application needs it. Several mechanisms exist to increase link reliability. A survey is given in [16]. One of the mechanisms used to improve link quality, by trying to eliminate or minimize interference, is channel hopping. It is a diversity technique that can help prevent external interference and multipath fading [17]. Channel hopping technique is used in several industrial 802.15.4-based [18] standards such as WirelessHART, ISA100.11a and IEEE 802.15.4e (Time Slotted Channel Hopping (TSCH) mode [29]). IEEE 802.15.4e is a MAC amendment of the existing standard 802.15.4-2006 designed for low power and low bandwidth reliable communication in industrial environments.

Existing industrial wireless technologies such as WirelessHART and ISA100.11a use a centralized network management approach. While a centralized approach can generate optimal results for static networks, it has several drawbacks. Firstly, the network manager is prone to a single point of failure. In case of failure or network partitioning, nodes that do not have access to the network manager are left without management functionality. Secondly, the centralized approach incurs a high communication overhead and latency for exchanging management traffic. Lastly, they cannot cope with network dynamicity in a timely manner. These problems are exacerbated as the network scales up. We show in this chapter that these problems are significant and we demonstrate how they can be solved.

This chapter presents a Distributed Management Scheme for Real-time applications (D-MSR) that is built for wireless industrial automation. Using a distributed approach, D-MSR could address the issues of high throughput and reliable communication as well as real-time requirements, while achieving

higher efficiency in network management in terms of delay and overhead. Issues such as node joining, reserving communication resources for exchanging management messages, constructing end-to-end connections between sensors and gateway/actuators for addressing real-time requirements, handling of network dynamicity such as node or edge [1] failures, and data delivery in case of lossy networks, are all addressed by D-MSR.

According to our knowledge, this is the first distributed management scheme based on the IEEE 802.15.4e standard (TSCH), which supports the whole protocol stack and manages the nodes in different phases, from joining to publishing the sensor/process data in the network. Related work mainly focused on the data link layer that provides data delivery service in a timely and reliable manner in multi-hop wireless networks, which are discussed in Section 4.2.2.2.

To address all the listed issues, we define different mechanisms and modifications in different OSI layers. In the data link layer, we define a two-hop neighborhood schedule-matrix that is used to construct a communication schedule between different pairs of network devices [2] in a distributed manner. In addition, two modules are defined in the upper data link layer: *neighbor connection manager* and *D-SAR*. The neighbor connection manager defines initial communication links between each node and its neighbors. Therefore, the upper layers can use these primary links to communicate with a particular neighbor. In order to reserve communication resources and provide real-time communication between two end-points based on the required bandwidth, we use the D-SAR signaling protocol [22]: a Distributed Scheduling Algorithm for Real-time applications based on concepts derived from Asynchronous Transfer Mode (ATM) networks [36]. The distributed nature of our resource reservation scheme, makes it feasible to change the reservation based on possible changes in the network connectivity, caused by the interference and dynamic link quality between the devices, in a timely manner. This capability together with the Clear Channel Assessment (CCA), re-transmission and channel hopping schemes in the data link layer, provide reliability in the network. As a response, D-MSR can address both the real-time and reliable communication requirements in a harsh industrial environment.

In the network layer, we use RPL (Routing Protocol for Low power and Lossy Networks [15]). In the transport layer we define, the *end-to-end connection manager* that establishes connections to either enable management communications (e.g., network layer control messages between network devices and gateway) or

[1] In this chapter "edge" means a node-to-node connection in the network layer. [2] The terms "network device" and "node" refer to a field device, such as a sensor and actuator, as well as router that improve network connectivity.

sensor/process data communications, through the D-SAR signaling protocol. The sensors publish periodic data to actuators (the term "sensor to actuator", "peer-to-peer" and "point-to-point" communication are used interchangeably) for process control applications, or to gateway for monitoring applications. In case of node or edge failures, the end-to-end connection manager releases the previously allocated communication resources along the old path, reserves new resources [3] and establishes a new connection between the pairs through the new path, by applying the D-SAR signaling protocol.

We compare via simulation the performance of D-MSR with that of WirelessHART (given the similarities between WirelessHART and ISA100.11a, the same result can be obtained by comparing D-MSR with ISA100.11a) in a typical industrial environment with high packet losses. We evaluate the end-to-end data delivery delay and compare the communication schedule and network throughput of D-MSR with that of WirelessHART. Furthermore, we evaluate the relationship between the packet delivery ratio and increased internal and external interference in the network. We show that in case of extensive external interference, D-MSR requires less time to reach a stable data delivery ratio value in comparison with WirelessHART. We compare the power consumption in the D-MSR network with that of WirelessHART. We show that by applying D-MSR, we can achieve higher efficiency in network management in terms of latency and overhead during node joining, resource reservation, end-to-end connection establishment, and when coping with dynamic situations (e.g., node or edge failure).

Section 4.2 describes D-MSR protocol stack architecture. Section 4.3 provides details about the functional description of D-MSR algorithms in different protocol layers. We provide details on the different phases of a network node from joining to publishing its sensor data, in Section 4.4. Section 4.5 elaborates on performance evaluation for real-time communication schedule construction, network throughput, data delivery in case of lossy networks, and management efficiency (in terms of delay, communication overhead), by comparing D-MSR with WirelessHART performance. Finally, Section 4.6 concludes the chapter and summarizes our future research in this area.

4.2 D-MSR Protocol Stack Architecture

In WirelessHART and ISA100.11a, a central network manager schedules all the network communications, constructs all the routes, and establishes end-to-end

[3] In this chapter "resources" means communication resources in the network

connections in the network. The protocol stack of WirelessHART, the connection between tables in different layers, and the managing procedures are shown in Figure 4.1(a). The network manager configures the communication tables in the data link layer and the routing table in the network layer through the system manager module implemented in each device. WirelessHART uses graph routing as well as source routing [13] in the network layer and use the Route Table and Source Route Table.

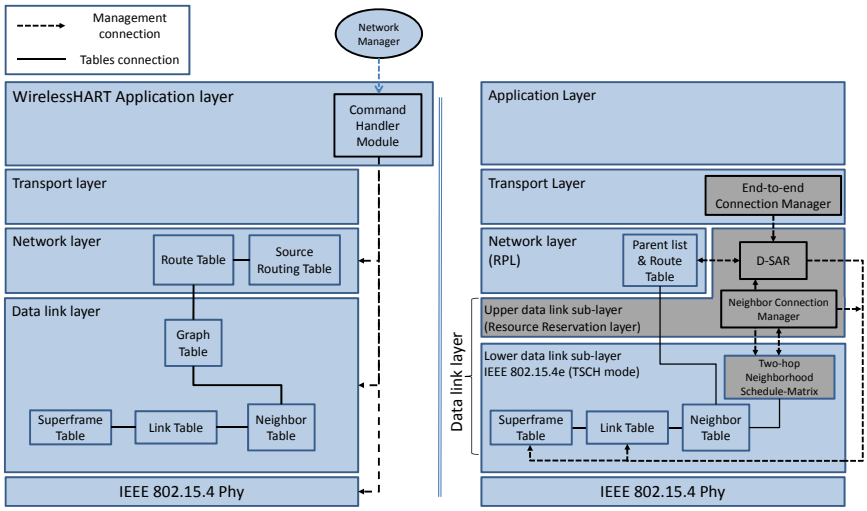


Figure 4.1: WirelessHART (a) and D-MSR (b) protocol stack

In D-MSR the network setup is performed in a distributed manner. This requires the implementation of various mechanisms in different layers. The D-MSR protocol stack is shown in Figure 4.1(b), in which the new sub-layers, modules and tables are displayed in a different color. The data link layer consists of two sub-layers: the lower and the upper data link sub-layer. In the lower data link sub-layer, we use the IEEE 802.15.4e (TSCH mode) standard, after having modified it to fit our requirements. A two-hop neighborhood schedule-matrix is added in this layer in order to schedule interference-free communications in the network. The modification details are discussed in Section 4.2.1. The upper data link sub-layer (the resource reservation layer)

supports several features and functionalities which are normally data link layer functions, but are not currently included in the lower data link sub-layer. In this sub-layer, we implement *D-SAR* and *neighbor connection manager* modules that configure locally the communication tables in the lower data link sub-layer. These two modules use the information provided by the schedule-matrix to construct interference-free schedules in different network operation phases. These two modules are discussed in more detail in Section 4.2.2. The end-to-end connection manager module is implemented in the transport layer. This module establishes the end-to-end connection through the D-SAR protocol. The modifications carried out in the lower data link sub-layer, the upper data link sub-layer, routing layer, and transport layer, as well as the ways in which they can work together, are discussed in Sections 4.2.

4.2.1 Lower Data Link Sub-Layer

In the centralized approach the network manager constructs the communication schedule in line with the network devices requirements based on the global knowledge it has obtained from the network. For instance, the network manager in WirelessHART maintains a global schedule-matrix to keep track of the timeslot-channel cell usage by the network devices. Allocation of an interference-free cell to one pair of neighbor devices is feasible since the network manager manages the usage of that cell by any other pairs (the term "interference" refers to the "internal interference" caused by the concurrent transmissions in the same channel in the network). In addition, the network manager avoids spatial reuse of that cell in the network. However, in the distributed approach we need a distributed management scheme to avoid allocating the same cell to another interfering pair of devices, either in the network or neighborhood. The interference models can generally be classified into: (1) *physical* and (2) *protocol* interference model [66, 67]. In the physical model, the feasibility of an interference-free communication is determined by the signal-to-interference ratio (SIR) of a receiver. In the protocol model, the feasibility of an interference-free communication is determined based on graph neighborhood relationship. In this chapter, conform the protocol model, a node uses information about the allocated cells in its two-hop neighborhood to reserve interference-free cells, after which it will monitor the status of its scheduled cells to guarantee interference-free communications. To this end, a two-hop neighborhood schedule-matrix [4] is defined in the lower data link sub-layer, in which

[4] The terms "schedule-matrix" and "two-hop neighborhood schedule-matrix" are used interchangeably

Table 4.1: List of additional information included in the advertisement payload

Additional Information	Description
Link table information of the advertiser	Used by receivers to construct the schedule-matrix (Discussed in Section 4.3.1)
List of advertisement cells used by the advertiser and its neighbors	Used by receivers to select a free advertisement cell (Discussed in Section 4.3.1)
List of free timeslots of the advertiser	Used by receivers to define initial communication links (Discussed in Section 4.3.2)

each node maintains the current usage of its two-hop neighborhood cells. Each entry in the schedule-matrix represents the cell usage at that timeslot on that channel and is specified by the node addresses of the scheduled link. In order to establish initial links and to enable further communication between neighbors in different network operation phases, the neighboring nodes need to find the same unused cell in their schedule-matrices. The procedure for constructing and updating the schedule-matrix are discussed later on in Section 4.3.1.

In D-MSR, we use an idle listening to update the schedule-matrix in each node. The nodes listen to their one-hop neighbors advertisements to update their schedule-matrices. To this end, the advertisement [5] also includes additional information about the subset of the advertiser link table (i.e., node address, timeslot, channel offset, and superframe ID) that are used by the receiver to construct and update its schedule-matrix. Furthermore, in TSCH it is assumed that the network manager schedules the advertisement links between the advertiser and its neighbors. In order to assure that in D-MSR the node can hear their neighbors advertisements, we modified the TSCH matrix in the lower data link sub-layer by defining two periods in the superframe; the *advertisement period* and the *data communication period*. In the advertisement period, nodes either send their advertisements or listen to their neighbor advertisement. No further communication links are scheduled allowing for more data sharing between the nodes in the advertisement period. The additional information that is included in the advertisement is listed in Table 4.1. In the data communication period, communication schedules are reserved to enable the communication between neighboring nodes. Figure 4.2 shows this setup. The figure illustrates a superframe with a length of 250 ms consisting of 25 slots.

In the *advertisement period*, the nodes can choose a free advertisement cell in channels 15, 20, and 25 (these three channels do not overlap with any of

[5] In TSCH nodes broadcast advertisements to enable network formation and to exchange timing information

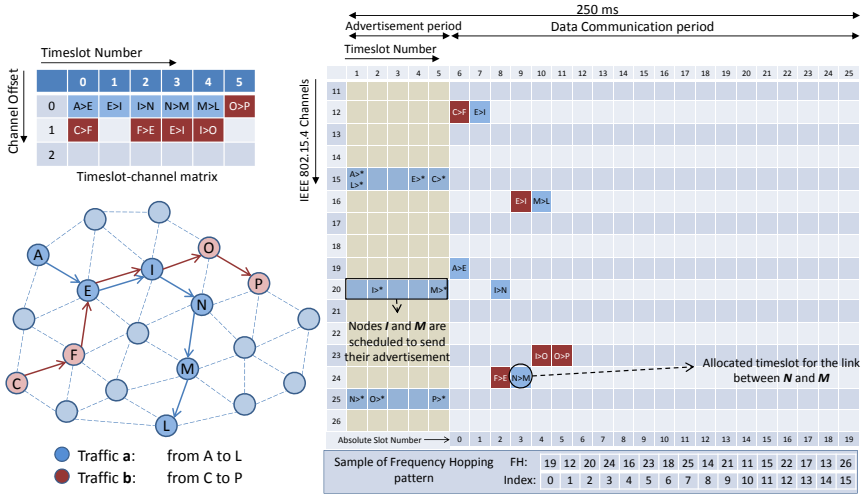


Figure 4.2: Modified Superframe

the three common IEEE 802.11 channels. Therefore, less interference occurs in these channels. A similar concept is used in the ISA100.11a standard, in which these three channels are designated as slow hopping channels for purposes such as neighbor discovery) to send the advertisements. We limit the number of channels in that period to three, in order to facilitate neighbor discovery and data sharing during joining. If a node chooses a timeslot to send the advertisement, the node will transmit an advertisement in the assigned channel most of the time. If not, it listens in a randomly selected channel (after having chosen from three advertisement channels) to receive other neighbors advertisements. However, if a node is not supposed to send the advertisement in a timeslot in the advertisement period, the node will once again listen in a randomly selected channel. The procedure of selecting a free advertisement cell will be discussed in Section 4.3.1.

In the data communication period, D-MSR schedules interference-free communication links between the neighboring nodes. For example, as is shown in Figure 4.2, traffic 'a' and 'b' are transmitted from node A and C toward node L and P respectively. The scheduled communication for these traffic flows are shown in the slot-channel matrix in the top left of Figure 4.2. Each time a scheduled communication is going to occur on a link, both sides of the link calculate

the radio channel of the communication by taking *"(Absolute Slot Number + Channel offset) % Number of channels"*. For instance, nodes I and N that select the timeslot 2 and channel offset 0, follow the frequency hopping pattern with that offset in the data communication period and will use channel 20 in that timeslot, as is shown in the data communication period (blue cells for traffic 'a' and red cells for traffic 'b') of the superframe. The procedure of scheduling the communication links (i.e., filling the link table and superframe table) in the data communication period are handled by the D-SAR and neighbor connection manager modules in the upper data link sub-layer, which are discussed in Sections 3.2.

4.2.2 Upper Data Link Sub-Layer (Resource Reservation Layer)

To enable the initial communication between two neighbor nodes (that can be used by the routing layer), these nodes should agree on the same link (timeslot and channel offset). Furthermore, based on the traffic that passes through this edge, more links need to be reserved to enable real-time end-to-end connections. In the centralized approach, the network manager schedules the initial communication resources as well as the required resources for further communications and fills in the data link layer communication tables in each network device based on those schedules. However, TSCH does not describe any distributed mechanism, by which either the initial communication links for neighbor nodes or more communication resources, for real-time end-to-end communications, can be allocated. For this reason, we define an upper data link sub-layer (resource reservation layer) on top of the data link layer, to configure the data link layer communication tables and to schedule the communications between neighbors.

Two modules are defined in the upper data link sub-layer: *neighbor connection manager* and the *D-SAR* module. The neighbor connection manager allows the TSCH MAC protocol to be glued onto the higher layer (routing layer), besides providing initial neighbor nodes communications. The D-SAR module reserves communication resources along the path in different phases of the network operation to enable real-time end-to-end connection either for management traffic purposes or to sensor/process data traffic. As is shown in Figure 4.1(b), neighbor connection manager and D-SAR modules configure the data link layer communication tables (the link table and superframe table), to allocate or release the communication resources. The remainder of this section focuses on the neighbor connection manager and D-SAR module respectively.

4.2.2.1 Neighbor Connection Manager Module

TSCH does not describe how the communication links should be constructed to enable initial communication of a node with a particular neighbor. However, the next upper layer (network layer) that resides on top of TSCH, assumes that nodes are capable of communicating with all their neighbors. In response, the neighbor connection manager (in the upper data link sub-layer) defines the initial communication links (one Tx-link and one Rx-link) between each device and its neighbors. This can be done by adding new links and superframes in the link table and superframe tables. The relation between the neighbor connection manager and communication tables in the lower data link sub-layer is shown in Figure 4.1(b).

In order to establish the initial communication links between neighboring nodes, they need to agree to communicate in a particular interference-free cell. To this end, a handshaking mechanism is needed between the new device and each of its neighbors to choose the common unused cell (i.e., timeslot number and channel offset). The details of handshaking mechanism are discussed in Section 4.3.2.

4.2.2.2 D-SAR Module

Real-time control applications require data to be transmitted over long distances through a multi-hop network in a reliable and timely manner. However, most recent studies [68, 69, 70, 71] on data link layer use the centralized resource reservation (scheduling) scheme to provide timely and reliable data delivery service. The centralized scheduling schemes have several disadvantages. They often perform poorly in terms of reaction time, as all updates need to be sent first to the base station for further processing. A distributed resource reservation algorithm is needed which would allow source nodes, based on the requirements of the application and traffic characteristic, to reserve network resources for its peer communications along their paths for addressing different QoS needs. Relevant techniques from other networking-related domains (e.g., Asynchronous Transfer Mode (ATM)) could potentially be adapted to develop solutions that are suitable for wireless sensor and actuator networks [16].

D-SAR is a distributed scheduling algorithm that is based on concepts derived from ATM networks. This is because the ATM signaling protocols [22] also address performance issues in terms of reliability and timeliness of packet delivery.

The D-SAR protocol is used to establish an end-to-end connection (for sup-

porting point-to-multipoint or point-to-point traffic) and to reserve the communication resources based on the traffic characteristics requested by the source node, along the path toward the destination in different phases of the network operation. These traffic flows can be either network management traffic (e.g., network layer control messages) or sensor data traffic that are published periodically by the sensor nodes toward actuators or gateway. The D-SAR module in the upper data link sub-layer receives the request for establishing a connection from the *end-to-end connection manager* in the transport layer. The D-SAR module in each device, reserves and releases the communication resources by modifying the link table and the superframe table in the lower data link sub-layer. The relation between the D-SAR module and the end-to-end connection manager and communication table is shown in Figure 4.1(b).

Before initiating the D-SAR protocol, the network is already established, all nodes have joined the network, the initial communication links have been established between neighbor nodes, and the routing layer has constructed the routes between network nodes. The details of the D-SAR protocol are discussed in Section 4.3.3.

At different phases of the network operation, the D-SAR protocol allocates or releases the communication resources (links and superframes), based on a request that may be initiated either from the upper layers in the stack or received from the other neighbors. In Sections 4.4.4 (Phase-4), 4.4.5 (Phase-5), and 4.4.6 (Phase-6) the details of these procedures are explained.

4.2.3 Routing Layer and Transport Layer

We use RPL in the routing layer. RPL is designed for Low power and Lossy Networks (LLNs), which consist of nodes with limited capabilities, such as processing power, memory, and battery power. RPL is defined for a network, in which nodes interconnections are lossy and the traffic rate is low [15]. These characteristics make RPL suitable for use in wireless industrial networks.

RPL is a distributed routing protocol that supports the up, down, and point-to-point traffic model by forwarding the packet to its selected parent from the parent list, based on the objective function (for example, by selecting the parent with the best Expected Transmissions values in the up direction) or by selecting a neighbor form the routing table as a next hop (in the down direction). The parent list and route table in the network layer, and their relationship to the neighbor table in the data link layer are shown in Figure 4.1(b). In the point-to-point traffic model, when a node (e.g., a sensor) needs to reach another node (e.g., a actuator), its packet travels in the "up" direction toward a common

ancestor and is then forwarded down toward the final destination. For example, as is shown in Figure 4.3, node 8 needs to communicate with node 11. The packet first travels "up" toward node 0. However, in the "up" route toward the root, the packet reaches node 2, which is a common ancestor between node 8 and 11. Node 2, which contains the destination address of the packet in its routing table, then forwards the packet toward node 11 through node 5.

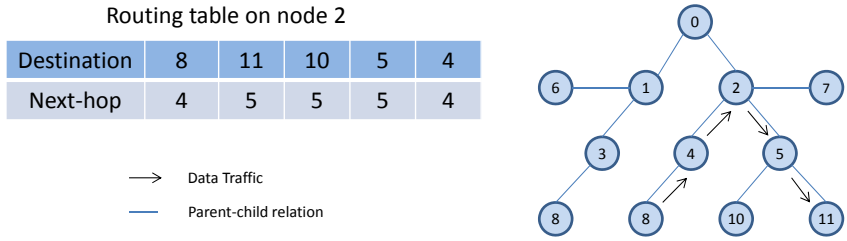


Figure 4.3: Point-to-point traffic model in RPL

In the transport layer, the *end-to-end connection manager* establishes the management connection (between new devices and the gateway) as well as an end-to-end connection (between sensors and gateway/actuators) through the D-SAR protocol. In the case of node or edge failure, the connection manager releases the previously allocated resources along the old path, and re-establishes a new connection by allocating new resources along the new path.

4.3 Functional Description of D-MSR Algorithms in Different Protocol Layers

In this section, we first illustrate the mechanisms used to select the advertisement cell and to construct schedule-matrices in the lower data link sub-layer. Next, we discuss the mechanism that is used to define the initial communication links with neighbors in the upper data link sub-layer. Finally, we explain the D-SAR protocol used to establish an end-to-end connection and to reserve the communication resources in the upper data link sub-layer.

4.3.1 Selecting Advertisement Cell and Constructing Two-Hop Neighborhood Schedule-Matrix

To let a new node choose the free advertisement cell in a distributed manner, the new device should listen to its neighbor's advertisement. The advertisement includes the advertisement cell numbers of a node and its neighbors. This effectively allows a receiving node to gather advertisement cell information about its two-hop neighborhood. The new device then chooses a free advertisement cell based on this information. A similar scheme is proposed to allocate the timeslot in a distributed manner in [72].

In the *protocol* interference model, the transmission on one edge (e.g., between node A and B) is interference-free and can only be activated in one timeslot-channel cell if there is no transmission on any edge that disturbs either A or B, as is shown in Figure 4.4(a). The conflicting edges (shown by black dashed lines) with edge (A, B) can be formulated, based on the [66] model, as follows:

$$ConflictSet((A,B)) = \{(C,D) \in E | [\{C,D\} \cap \{A,B\} = \oslash] \wedge [\{A,B\} \cap (R_C \cup R_D) \neq \oslash]\} \quad (4.1)$$

ConflictSet denotes the set of conflicting edges with the edge (A, B). The set of all edges in the network is denoted by E while R_C denotes the set of nodes that are possible receivers of node C. In addition to the ConflictSet, other edges (shown by blue dashed lines) that are sharing a node with edge (A, B) cannot be scheduled in the same cell. That is because we assume that each node has a single radio transceiver and cannot simultaneously receive and transmit. In a realistic setting, the interference and transmission range of a node may not be equal. However, in D-MSR we assume, for simplicity, that the interference and transmission range of a node are equal. In case these ranges are not the same, considering an additional virtual edge representing the interfering edges [66] can be a possible solution. The details of virtual edge mechanism are discussed in Section 4.4.6.3.

Each node maintains a schedule-matrix to keep track of the current cell usage in its two-hop neighborhood, as shown in Figure 4.4(b,c) for nodes A and B. The schedule-matrix is constructed based on the link table information that the node collects from its one hop neighbor's advertisements. In the received link table information from one hop neighbor, the links between the one-hop and two-hop neighbors are included. Any two nodes that wish to establish an

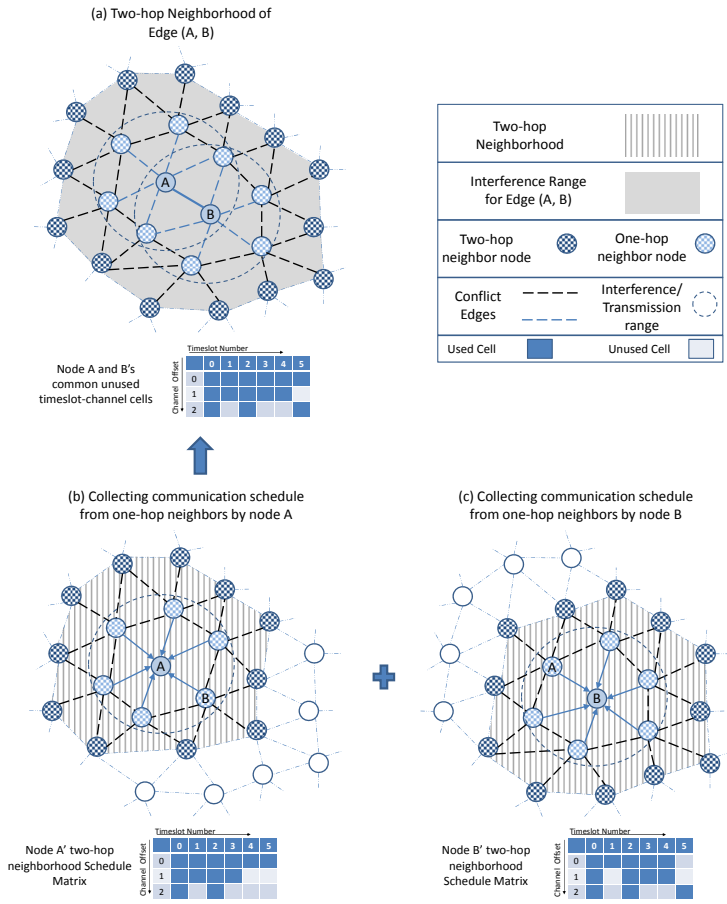


Figure 4.4: Selecting an interference-free cell on (a) edge (A, B) based on the constructed schedule-matrix of (b) node A and (c) node B

interference-free link with each other can negotiate based on their schedule-matrix and find a common cell that is not used by any of their possible conflict edges in their own two-hop neighborhood.

4.3.2 Defining Initial Communication Links with Neighbors

The idea of defining the initial communication links with neighbors is derived from [73]. In [73] the authors describe the algorithm that provides the initial communication link between a mobile node and its adjacent neighbors. Mobile nodes change their connectivity very rapidly as a result of which the reservation of communication resources and the provision of real-time communication between two end-points are not considered. However, by modifying this algorithm based on our requirements, the initial communication links between a node and all of its adjacent neighbors can be scheduled. In [73], while the nodes are trying to schedule the communication links with their neighbors, they choose a random channel offset for each link and use that channel for their further communications on that link. However, assigning a different channel offset to conflicting edges is not discussed in [73]; to handle internal interference, nodes need to ensure that while communicating nodes choose the same frequency, conflicting edges use different channels. Moreover, in [73] the advertisements are sent on channel 0 and all the neighbor nodes listen on channel 0 in their free timeslots to receive the advertisements. As the nodes schedule fills up, they spend less time listening on channel 0 for advertisements. This means that nodes with more busy schedules have difficulties adding more bandwidth. D-MSR allows for more data sharing between nodes by considering the special period in each superframe for sending advertisements.

We define five states: "*Aloha*", "*Transmit Connection Request*", "*Receive Connection Request*", "*Transmit Data*", and "*Receive Data*" for each timeslot, as in [73]. The default state for all the timeslots in the data communication period is Aloha.

Figure 4.5 illustrates different states of a sample timeslot in the data communication period. At the beginning of each superframe, each node sends an advertisement in the scheduled advertisement cell in the advertisement/broadcasting period. This advertisement includes free timeslots, i.e., the timeslots with Aloha state in the data communication period. To assure interference-free communication, the advertisement suggests for each free timeslot an unused channel offset chosen from the free cells in the timeslot column at the schedule-matrix. After sending the advertisement, the advertiser changes the state of these free timeslots from Aloha to Receive Connection Request state, and listens for a potential Connection Request from the neighbors in the suggested channel. A neighbor node that receives the advertisement, checks whether it has any timeslot with Transmit Data state with the advertiser or not. If not, the neighbor tries to find a common unused timeslot-channel cell with the advertiser. Once found, it converts the selected timeslot state from Aloha to Transmit Connection

Request. The neighbor sends a Connection Request to the advertiser in the selected timeslot-channel entry. By receiving the Connection Request packet, the advertiser changes the state of that timeslot from Receive Connection Request into Receive Data and sends the acknowledgement of receipt to the neighbor. Upon receiving the acknowledgment, the neighbor changes the state of the selected timeslot from Transmit Connection Request to Transmit Data. If no Connection Request is received by the advertiser, the state of that timeslot is changed to Aloha. This procedure continues until the new node has established one timeslot with Transmit Data state and one timeslot with Receive Data state with all of its neighbors. Subsequently, a new node writes interference-free links in the communication tables, one Tx-link and one Rx-link for each of its neighbors. The channel offsets and timeslots of these links are set to the negotiated timeslot-channel entries, and the typical superframe (the length of the initial superframe is assumed to be 2 seconds) is added to the communication tables.

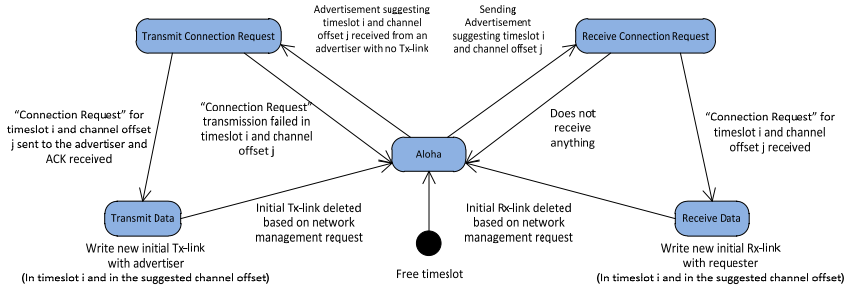


Figure 4.5: Different state of sample timeslot in the data communication period

4.3.3 D-SAR Protocol

The end-to-end connection manager in the transport layer of a source node, which intends to establish a connection, sends the connection-request to the D-SAR module in the stack, including the connection parameters such as a destination address, traffic/connection ID, connection priority (we use the same priority of data as defined in the WirelessHART protocol for exchanging the management, sensor data, alarm, or normal packets), communication type (periodic or non-periodic), and a requested publishing period. In this chapter, we assume the prevalence of periodic data traffic between sensors and actuators.

The D-SAR module at the source node initiates the procedure by sending a *Setup* message to the next hop toward the destination along the route defined by the routing layer. The Setup message includes parameters such as a list of suggested common unused timeslot-channel cells for further communication with the next hop, a destination address, traffic ID, timeslot-channel cell selected on previous hop (the information about the timeslot-channel cell selected by the previous hop, is used by the next hop in order to minimize the end-to-end delay), and a requested publishing period. The sender selects these common unused cells based on the received information about the next hop link table (by listening to the next hop advertisement) and its own schedule-matrix. The receiver of the Setup message then performs a check of its available communication resources. The receiver checks whether any of the suggested cells are unused in its own schedule-matrix with the requested publishing period. It also checks if there are unused cells with the requested publishing period to communicate with the next hop. If the required resources are available, the receiver chooses one cell from the suggested free cells and allocates the requested communication resource based on the requested publishing period of the traffic by writing a new link and (if needed, new) superframe in the related tables in the data link layer. The receiver will then respond by sending the *Call Proceeding* message that includes the chosen cell. In the next step, the receiver (intermediate node) forwards the Setup message toward the destination node with some delay. This delay enables the neighbors to update their schedule-matrices based on this new reservation that will be published in advertisements, thereby avoiding conflicts over resource reservation. This process continues until the destination node receives the Setup message as shown in Figure 4.6(a). However, at any intermediate node the receiver of the Setup message can refuse the connection request with a *Release Complete* message if it is unable to accommodate the new connection as shown in Figure 4.6(b).

The destination node can either accept or decline the new connection request from the source node by sending the *Connect* message or Release Complete message. This Connect message traverses along the multi-hop network back to the source node. All the temporary communication resources, which are reserved during the Setup message exchanging, are switched to permanent reservation. This two-step reservation is performed to ensure that timeslot reservations are not carried out should the connection request be unsuccessful.

After establishing the connection and during the network operation, either the source node (e.g., because the connection has expired or is no longer required), the intermediate node (e.g., because of node/edge failure, changing the route or detecting the conflict in the reserved resources), or the destina-

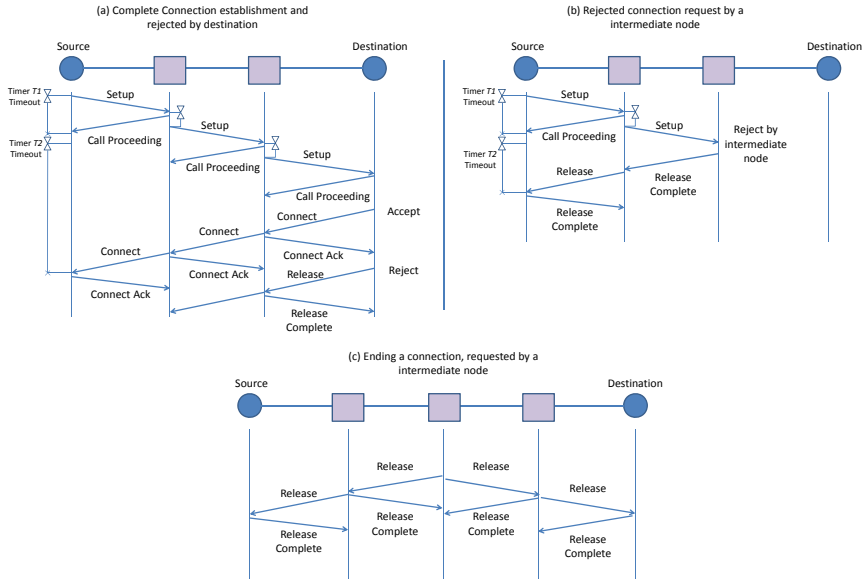


Figure 4.6: Overview of connection establishment protocol

tion node may wish to end the connection. The node that wishes to end the connection transmits the *Release* message toward the end(s) of the connection. The procedure of ending a connection by an intermediate node is shown in Figure 4.6(c). The receiver of the Release message deletes the communication schedule established with the sender and a Release Complete message is sent to the sender. The communication schedule is specified by the traffic ID. Next, the receiver of the Release message forwards the Release message to the next hop in the route toward the end-point of the connection. Upon receiving the Release Complete message from the next hop, it will then delete the communication schedule constructed with the next hop, which is specified by that traffic ID. This process continues until the Release message reaches the end-point of the connection. This procedure ensures that all nodes along the route release all the resources previously allocated to the connection. In this case, the D-SAR module in each node deletes the related links and superframes from the communication table. The details of the D-SAR protocol for the source, the intermediate, and destination node are provided in [74].

4.4 D-MSR Management Phases

In this section we discuss the different management phases, which guide the new node from startup to the moment the node starts to publish/subscribe the periodic sensor data in the network. The node operation state machine is shown in Figure 4.7.

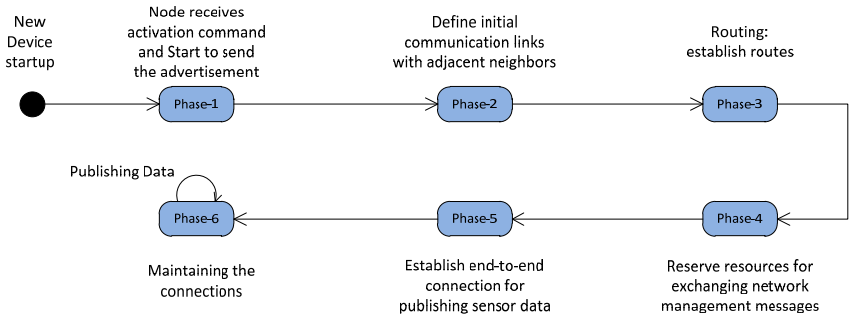


Figure 4.7: Different states of a node operation in the network

After a new node startup, in Phase-1 the node receives the activation command from the neighboring advertiser and starts to send the advertisement. In Phase-2 the initial communication resources between the node and its adjacent neighbor are allocated, by which the routing layer in Phase-3 can establish the routing graph. In Phase-4, the required communication resources should be allocated in the network to exchange the management messages in the routing layer. After construction of the routing graph and allocation of management resources, the end-to-end connection can be established between the sensors and actuators/gateway to publish the sensor data toward the destination(s) which is done in Phase-5. At the network setup stage each node goes through these phases. This procedure continues until all the devices have joined the network and started the operation. During normal operation of the network, in Phase-6, the D-MSR maintains the end-to-end connections by coping with dynamicity, by handling the resources reservation conflict, and by coping with internal and external interference. The following sections discuss these phases in more detail.

4.4.1 Receiving an Activation Command and Starting to Send the Advertisement (Phase-1)

The new device that intends to join the network listens on a physical channel for a period of time and then continues on the next channel, until all the channels have been scanned. The new device selects the best advertiser/candidate according to predefined criteria and sends the join request to the selected advertiser. In this work we select the advertiser according to the Link Quality Indicator (LQI) or Received Signal Strength Indicator (RSSI) of the received advertisement, although other criteria can be easily added. The advertiser sends the join response/activation command to the new device, upon acceptance (e.g., if the advertiser can still admit new devices). Sending the join request and receiving the join response procedure is implemented using the IEEE 802.15.4e standard. The joining procedure of a new device is shown in Figure 4.8(a).

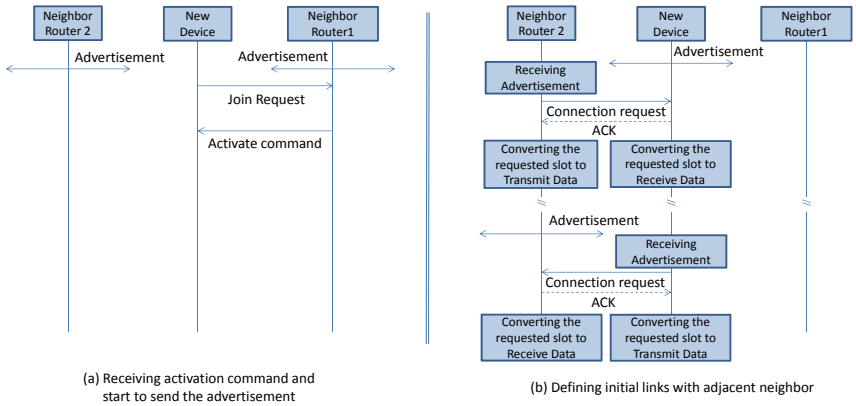


Figure 4.8: Receiving activation command (a) and defining initial communication links with neighbors (b)

Upon receiving the activation command, the new device starts to send the advertisement. However, before starting to send the advertisement, the new device should choose a free advertisement cell by listening to its neighbors advertisement (discussed in Section 4.3.1). The new device can choose a free advertisement cell based on this information and then start to send the advertisement in the advertisement/broadcasting period.

4.4.2 Defining Initial Communication Links with Neighbors (Phase-2)

After the new device joins the network, it needs to find the route toward the other nodes in the network or the gateway. The neighbor connection manager module in each network device, uses a handshaking mechanism (explained in Section 4.3.2) in order to define one Tx and one Rx link with each of its neighbors. Those links and a typical superframe will be added in the data link layer communication tables. These links enable a node to communicate with all its neighbors. Afterwards, the routing layer can be run to find the path between the endpoints. This procedure is shown in Figure 4.8(b).

4.4.3 Constructing the Routes (Phase-3)

In this phase, the routing layer finds the routes between the endpoints. In D-MSR we have used RPL in the routing layer. RPL specifies how the new device finds a path toward the gateway. By generating the RPL control messages, the routing entries in the intermediate nodes will be constructed as well as a complete path toward the new device. Several control messages, e.g., DAO (Destination Advertisement Object control message is used to construct routes to the other intermediate or leaf nodes) message, are forwarded through the network periodically to maintain and update the "up" (multipoint-to-point) and "down" (point-to-multipoint) routes.

4.4.4 Reserving Management Resources (Phase-4)

In this phase, the node reserves resources for exchanging network management messages. Once the node joins the network, in Phase-2 the initial communication links to adjacent neighbors are constructed and then in Phase-3 the routing layer constructs the "up" and "down" routes. In this phase, it is necessary to reserve the communication resources by which the routing layer control messages can be forwarded to the destination along the path. To reserve the management resources through the "up" path, each node runs the D-SAR signaling protocol to allocate the required resources based on the DAO messages rate (which is defined in the routing layer). Similarly, to reserve the resources through the "down" path, the root runs the D-SAR signaling protocol toward the new nodes.

In a centralized approach, such as WirelessHART or ISA100.11a, a different procedure is defined to receive the join request from the new device, send the activation command, construct the new graphs for the new device, and reserve

the management resources (e.g., management superframes and links). In these standards, the join request will be forwarded toward the network manager via the proxy device, and the network manager who has received the join request will use its centralized algorithm to allocate the management communication resources (such as graphs, superframes, and links). In the centralized approach, a join response/activation command is sent to the device after all necessary communication resources for exchanging the management messages have been configured and reserved along the path. The joining sequence of a new device in WirelessHART network is discussed in [20].

In D-MSR we consider the node as having joined the network, after it received the activation command from the neighbor advertiser, started to send the advertisement periodically (Phase-1), defined the initial communication links with adjacent neighbors (Phase-2), constructed routes to the other nodes (Phase-3), and reserved the communication resources to exchange the management messages (Phase-4).

4.4.5 Establishing an End-to-End Connection for Periodic Sensor Data Communication (Phase-5)

Having allocated the initial resources, as well as the management resources, the focus of this phase is to establish an end-to-end connection between a sensor and an actuator or a sensor and the gateway for transporting the application data. *Control in the field* (i.e., closed-loop control through a peer-to-peer communication between a sensor as a publisher and an actuator as a subscriber. This is part of traditional Fieldbus technologies) is important for process control applications (see Class 1 in Table 1.1). WirelessHART networks support peer-to-peer communication between sensors and actuators only if the traffic is routed via the gateway. This is required from WirelessHART's security mechanism to prevent potential safety threats resulting from undetected and unmonitored communications [13]. ISA100.11a addresses control in the field by providing a secured peer-to-peer communication. D-MSR addresses real-time communication between sensors and actuators (providing control in the field) as well as between sensors and the gateway.

As we focus on applications that require constant data traffic rates, D-MSR allocates a virtual circuit for each traffic flow. This implies that the resources reserved for each end-to-end connection depend on the traffic characteristics requested by the source node. The source node initiates this phase by sending a *Setup* message as was shown in Figure 4.6. The format of this message is

similar to the Request Service in WirelessHART and the Contract Request in ISA100.11a.

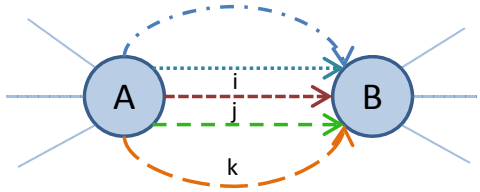
In WirelessHART, if the sensor node needs to have a connection with another device, which can be an actuator or gateway, it will send the Request Service to the network manager with specified bandwidth and latency characteristics. The network manager needs time to schedule new communications along an uplink graph from the sensor to the gateway, and from the gateway to the actuator along a downlink graph. It will then reply to the requesting node. The process of asking for more communication resources is discussed in more detail in [13]. However, unlike in ISA100.11a and WirelessHART, which both send the request to a centralized network manager, in D-SAR the source node sends the Setup message toward the destination node along the route defined by the routing layer in a distributed way.

The traffic ID parameter, which is included in the Setup message, is used to specify the allocated resources for that traffic ID. For example, in case of releasing the specific connection resources, the traffic ID is used to identify the related communication resources that are allocated for that connection. However, to allow for the efficient utilization of each link during the normal network operation, they are shared, upon their allocation, between multiple traffic flows rather than assigned specifically to one particular traffic flow. This means that the communication resources, which are reserved for initial communications, management communications, or different end-to-end connections between sensors and actuators, are shared between different traffic flows. For example, let us consider nodes A and B in Figure 4.9. Five links are established between the two nodes in different phases (e.g., link (II) that belongs to the superframe with 2 s length is established in Phase-4 for exchanging management messages, and link (III) that belongs to the superframe with 250 ms length is established in Phase-5 for forwarding traffic ID i). Using the ATM networking concepts, management traffic, traffic ID i, traffic ID j, and traffic ID k are allowed to use all the defined links between node A and B during the normal operation of the network.

4.4.6 Coping with Dynamicity, Reservation Conflict and Interference in the Network (Phase-6)

4.4.6.1 Coping with Dynamicity in the Network

In order to cope with network dynamicity, such as node or edge failure, the connection manager in a node (i.e., incident nodes of the broken edge or adjacent



	Links ID:	Belongs to:	Established in phase:
	Link (I)	Superframe 2s	Phase-2; Initial Link
	Link (II)	Superframe 2s	Phase-4; Exchanging mng. messages
	Link (III)	Superframe 250ms	Phase-5; Forwarding traffic ID i
	Link (IV)	Superframe 4s	Phase-5; Forwarding traffic ID j
	Link (V)	Superframe 500ms	Phase-5; Forwarding traffic ID k

Figure 4.9: ATM concepts in link definition between nodes A and B

nodes of the failed node that are part of end-to-end connections), transmits *Release* message(s) toward the source node(s) or destination node(s) by applying the D-SAR protocol. In case of edge failure, the incident nodes of the broken edge (node A and B in Figure 4.10) transmit the Release messages, including the traffic ID information, toward the end-points of each connection that passed through the broken edge.

The process of releasing the reserved communication resources, which is identified by the traffic ID, is executed for each of the connections containing the broken edge. At this stage, all the resources previously allocated to that connection will be released and become free. This means that the related links and superframes are deleted from the communication tables of each device in the former route.

As Figure 4.11 illustrates, in case of node failure, the adjacent nodes which joining edges are part of an end-to-end connection, release the allocated resources by transmitting the release messages toward the sources or destinations of the end-to-end connections. Exchanging the *Release* and *Release Complete* messages and releasing the resources follows the same procedure of edge failure.

The routing layer repairs the former routes. Upon receiving the Release

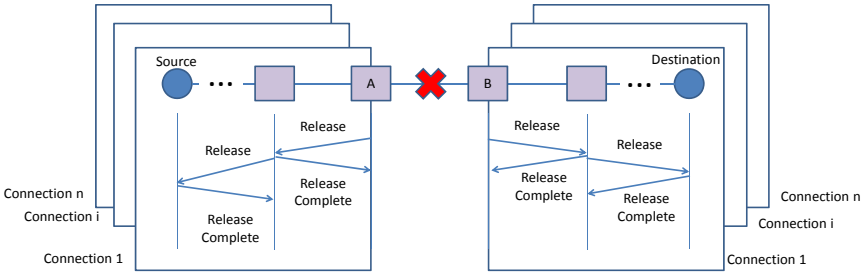


Figure 4.10: Releasing the allocated communication resources after edge failure

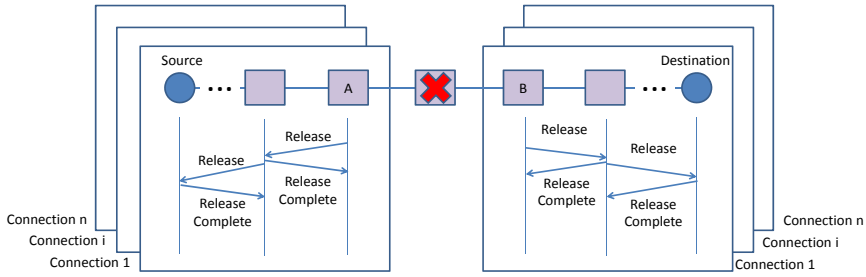


Figure 4.11: Releasing the allocated communication resources after node failing

message, the connection manager in the source node will re-establish a new connection and reserve the new resources along the new path once more by using the D-SAR protocol.

In case of a node or an edge failure in centralized approaches like WirelessHART or ISA100.11a, the failure should be reported to the network manager. Subsequently, the network manager establishes new routes, releases the previous communication schedule, and constructs new schedules.

4.4.6.2 Handling the Resource Reservation Conflict

In D-SAR protocol, the two nodes of an edge participating in end-to-end connections, negotiate to reserve a common unused timeslot-channel cell based on their

current two-hop neighborhood schedule-matrix. By considering the intentional delay before forwarding the *Setup* message, we allow their neighbors to update their schedule-matrix based on the new reservation. However, there is still a probability that a conflicting edge may also choose that cell, prior to receiving the new advertisements listing the changes in their neighborhood schedules. The nodes of the conflicting edges that have reserved the same cell should handle this conflict upon detection (the detection is done by observing the constant packet loss in that cell), by releasing the conflicting reserved resources. As a response, the end-to-end connection manager in the node transmits Release messages toward the end-points of the connection that include the conflicting cells. Figure 4.12 illustrates these two potential reservation conflicts scenarios. In the first scenario, when the Setup message (e.g., for traffic a) is being forwarded along the path, the same cell is chosen by edge (E, I) and its interfering edge (N, M). That is because node N did not receive the node I advertisement to update its schedule-matrix based on the new reservation on edge (E, I). This possible conflict is avoided in D-SAR protocol by the considered intentional delay in forwarding the setup message. The second conflict happens, when two setup messages (that belonged to two different end-to-end resource reservation) choose the same cell simultaneously in conflicting edges (I, N) and (O, P).

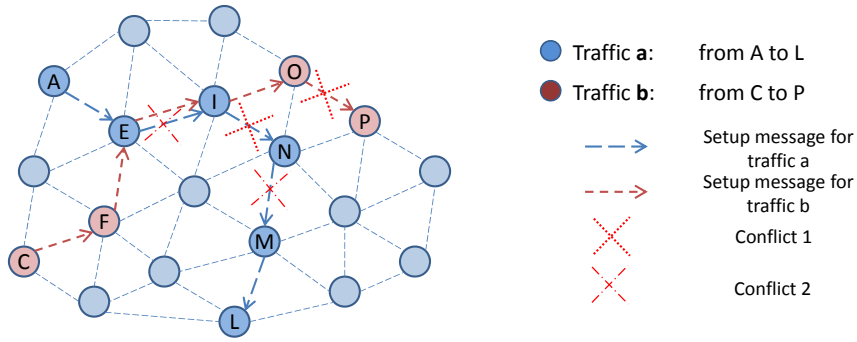


Figure 4.12: Resource reservation conflict sample

4.4.6.3 Coping with Internal Interference in the Network

In a realistic setting, the interference range of a node may be much larger than its transmission range. Concurrent transmission in the same cell may cause interference even when the edges are two hops away from each other. Figure 4.13 illustrates how the communication on edge (C, D) that is outside of the two-hop neighborhood of edge (A, B) interferes with edge (A, B). Thanks to the scheduled communications concept, internal interference caused by communications outside of the two-hop neighborhood, happens in specific timeslot-channel cells that can be recognized by (1) observing the constant packet loss in those cells after reservation or (2) by performing CCA before reservation. By considering the *virtual* links that represent the interfering links, adding these in the schedule-matrix and by subsequently avoiding to use those timeslot-channel cells, the internal interference can be solved in a distributed manner. In Section 4.5.5.1, we evaluate the effect of this scheme in improving the packet delivery ratio in case of internal interference in the network.

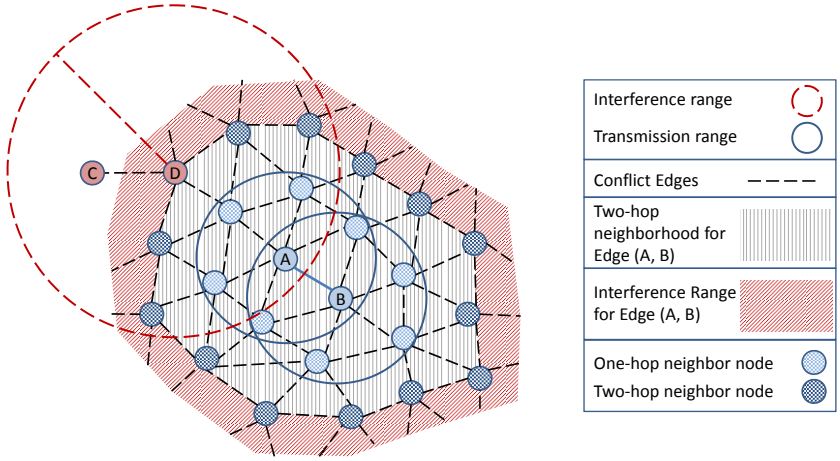


Figure 4.13: Internal interference caused by communication on edge (C, D) outside of the two-hop neighborhood of edge (A, B)

4.4.6.4 Coping with External Interference in the Network

In case of interference in the network, different edges may experience a different packet loss ratio. In centralized approaches like WirelessHART, each node periodically reports on the status of its communication with its neighbors to the network manager through a set of report commands. The network manager may re-construct new graphs, which include more reliable edges, based on the received reports from the network. It then releases former resources and constructs new communication schedule along these new graphs. These instructions will be forwarded to the network. This approach cannot cope with disturbance in large-scale networks in a real-time manner.

However, in D-MSR, the RPL uses best Expected Transmissions values (the expected number of transmissions required to successfully transmit and acknowledge a packet on the edge), as a metric, to find the best paths in case of interference. Subsequently, after choosing the new path, the previous resources along the old path are released, and the new communication resources will be reserved along the new path in a distributed manner. In Section 4.5.5.2, we compare the performance of D-MSR to that of WirelessHART in terms of the ability to provide reliable communication in case of interference in the network.

4.5 Performance Evaluation

This chapter has discussed the distributed management scheme ability to serve applications requiring a real-time and reliable communication as well as a high throughput. This section illustrates how these requirements are fulfilled. To this end, we first assess the end-to-end data delivery delay of D-MSR and WirelessHART. Next, we evaluate the communication schedules and network throughput. Following this, we assess the packet delivery ratio in case of internal and external interference. Furthermore, the power consumption in the D-MSR network and WirelessHART is being evaluated.

Finally, we evaluate the management efficiency of D-MSR in terms of delay and overhead during node joining, management resource reservation, end-to-end connection establishment, and coping with changes and disturbances in the network.

Table 4.2: NS-2 simulation parameters

Parameter	Value	Parameter	Value
Number of nodes	Gateway, two access points, 53 field devices	Radio range	15 meters
Simulation area	$100 \times 100m^2$	Frequency Band and channel	2.4 GHz, 11–26 channels
Placement	Regular distribution	Sensor traffic rate	1 per 2 seconds
Data rate	250 kb/s	Application traffic model	CBR

4.5.1 Implementation of D-MSR and WirelessHART in NS-2

We implemented the D-MSR protocol stack in NS-2. In the data link layer we implemented IEEE 802.15.4e (TSCH mode). In the routing layer we implemented RPL in NS-2.

We also implemented the WirelessHART protocol in NS-2 [20]. It is the first implementation that supports the WirelessHART network management algorithm as well as the whole protocol stack of the WirelessHART standard.

4.5.2 Simulation Model, Parameters and Network Topology

In the simulations we set a network area of $100m \times 100m$, the transmission range of 15 meters, and neighbors distance of around 10 meters. We use the *two-ray ground* model as a radio propagation model. The network consists of 46 wireless nodes that are evenly distributed in the simulation area. The network topology is shown in Figure 4.14. This regular topology helps to evaluate the behavior of D-MSR and WirelessHART more accurately. For instance, in Section 4.5.5.2, we can evaluate the effect of increasing interference regions on the data delivery ratio rather precisely, by controlling the number of edges that were affected by interference in each step.

The length of management superframes, which is defined to allow for the exchange of management messages, is set to be 2 seconds. All the obtained results are based on the 2 seconds management superframes. In addition, in D-MSR, the size of link table entries, which are included in the advertisement payload, may reach 400 bytes. In the simulations, we assume that these amounts of data can be compressed in the advertisement payload with a size of 100 bytes. The detailed parameters are presented in Table 4.2.

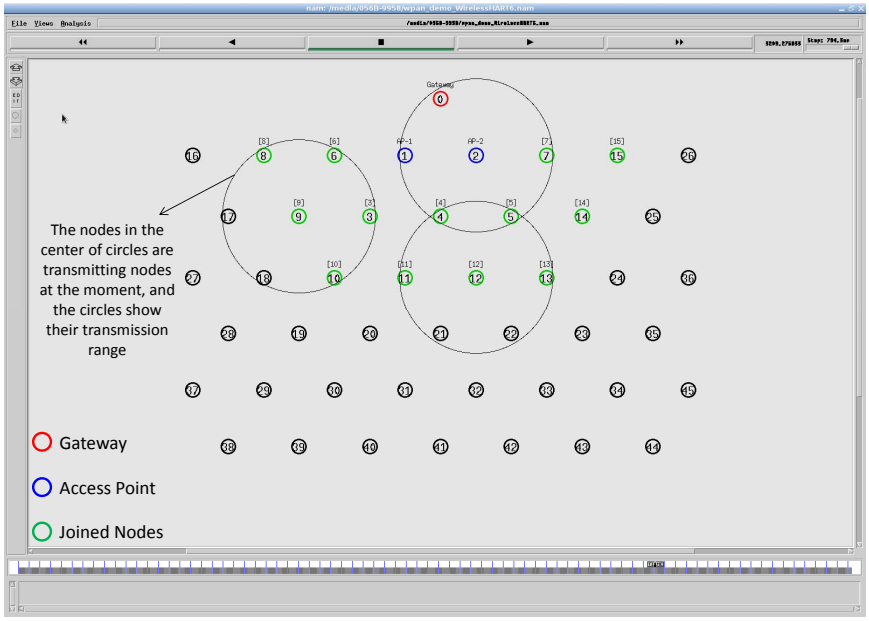


Figure 4.14: The network topology from animation tool of NS-2 simulator (nam)

4.5.3 Real-Time Evaluation

To evaluate the end-to-end data delivery delay, 29 pairs of sensors and actuators were considered in the network. These pairs are chosen in such a way that the total hop distances of the sensor to the gateway and of the gateway to the actuator are spread in different hop levels. In Figure 4.15 a sample of an end-to-end connection is shown between a sensor node (37) and an actuator node (45) based on WirelessHART and D-MSR network, respectively.

We evaluate the average end-to-end data delivery delay and the average number of hops that the received packets need to travel to reach their destinations through the 29 connections. The results are shown in Figure 4.16 for both D-MSR and WirelessHART. In this figure, we classified connections into five categories based on the total hop distance of sensor to actuator via the gateway. We forward the traffic (periodic sensor data) from sensors towards actuators, by employing the constant bit rate (CBR) traffic model in NS-2 for

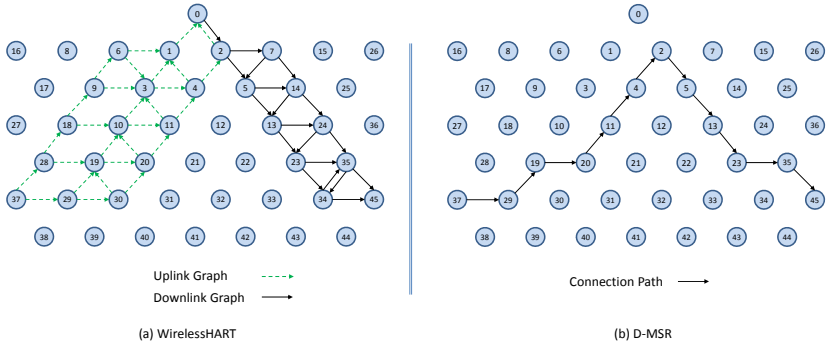


Figure 4.15: End-to-end connection between nodes 37 and 45 in WirelessHART (a) and D-MSR (b)

all end-to-end connections. The requested publishing period of the sensor data for all 29 connections is set to two seconds. Subsequently, communication resources are reserved to exchange sensor data messages between the sensors and actuators/gateway based on that period.

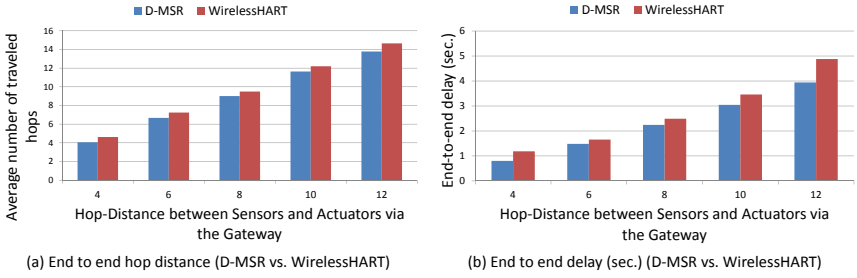


Figure 4.16: Average end-to-end hop distance (a) and delay (b) (D-MSR vs. WirelessHART))

The end-to-end delay in D-MSR is close to that of WirelessHART, which implies that D-MSR achieves similar results in addressing the real-time requirements during an operational phase. In those connections, in which the periodic sensor data packets have to travel more hops to reach their destination, more

end-to-end delay is expected.

4.5.4 Network Throughput

In this section, we compare the communication schedule and network throughput of D-MSR with those of WirelessHART. Figures 4.17 and 4.18 show samples of constructed schedules for 29 end-to-end connections with a publishing period of two seconds in WirelessHART and D-MSR respectively. In these matrices, the communication schedule reserved for transmitting either management traffic or sensor data are shown. Through different colors in each cell in the matrix, the number of edges re-using that particular timeslot-channel cell are shown. Figure 4.17 shows the global matrix [6] of the allocated timeslot-channel cells by the WirelessHART network manager[7]. In this scenario, the network manager schedules each communication in an interference-free cell and avoids the spatial reuse of any cell between different edges, except for during the advertisement period. Figure 4.18 shows the combination of all schedule-matrices in each node in the network, which represents the global schedule-matrix (the combination of superframes with a size of 25 and 200 timeslots) in D-MSR. The D-MSR matrix looks more dense with more unused cells. There are two reasons for this. Firstly, in D-MSR the nodes just keep track of current cell usage in their own two-hop neighborhood. This means that reuse of the same cell in different two-hop neighborhoods could occur. As is shown in Figure 4.18, a given cell may be reused by 10 edges in different neighborhoods. Secondly, since more edges are considered in the uplink and downlink graph, more communication schedules are constructed in the WirelessHART network.

In addition, we evaluate the network throughput of both D-MSR and WirelessHART in different network densities. We gradually increase the transmission range of nodes in five steps from 15 to 25 meters to provide a different network density from seven to 21 neighbors in the one-hop neighborhood. For each network density, to evaluate the reachable network throughput, we try to establish the maximum number of end-to-end connections between field devices. As the network density increases, more bottlenecks are observed and less end-to-end connections can be established. In D-MSR more end-to-end connections can be established thanks to the RPL in the routing layer which does not need to route the traffic through the access points. On the other hand, the implemented WirelessHART passes all the traffic through the gateway. Fig-

[6] The combination of superframes with a size of 25, 200, 400 and 800 timeslots. [7] The advertisement period is also considered for the WirelessHART network to ensure a fair comparison of communication resources.

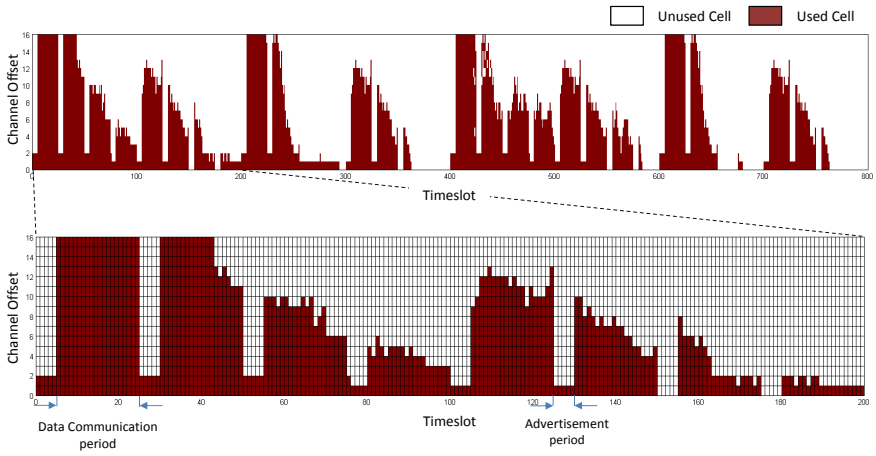


Figure 4.17: The global matrix of the current slot/channel usage for the sample WirelessHART network

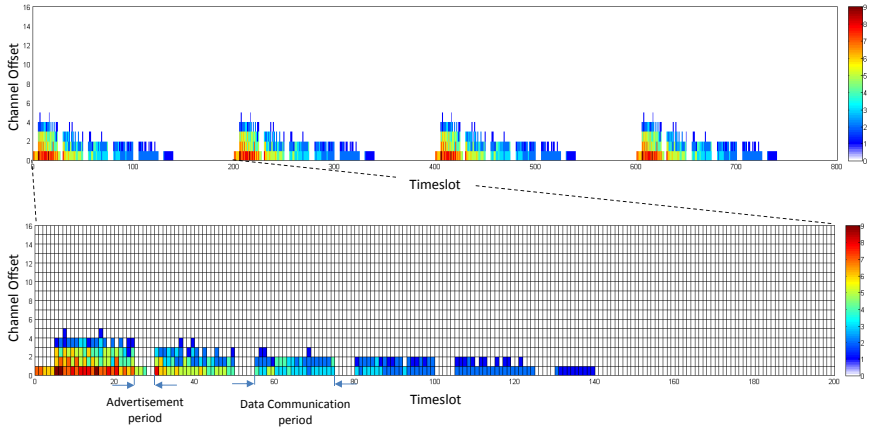


Figure 4.18: The combination of schedule-matrix of the current slot/channel usage for the sample D-MSR network

ure 4.19 shows the network throughput: the number of transmitted packets in

the whole network per second. As the network density is increased, the network throughput for both D-MSR and WirelessHART decreases, but less severely so for WirelessHART.

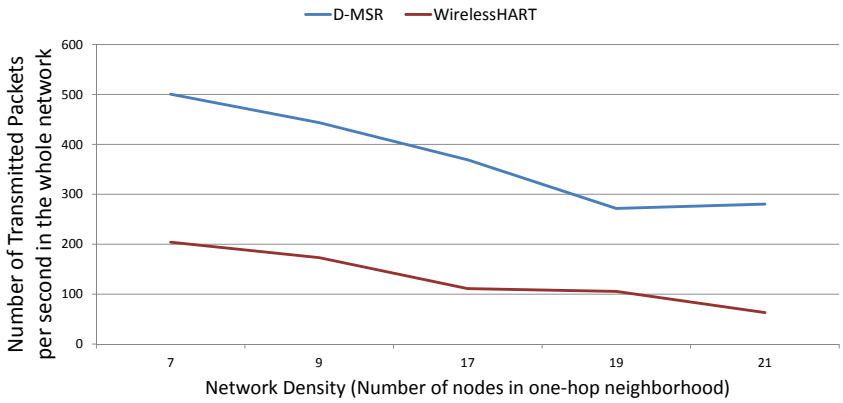


Figure 4.19: Network throughput (D-MSR vs. WirelessHART)

The spatial reuse of communication resources provides more network throughput for D-MSR than WirelessHART in the case of a sparse network. For example, when the number of neighbors in the one-hop neighborhood is seven, the network throughput is around 150% higher for D-MSR than WirelessHART. However, as the network becomes more dense, less disjoint two-hop neighborhoods can be observed. Consequently, less communication resources can be reused, which results in less network throughput difference. The communication schedule of D-MSR and WirelessHART for each of those five densities are shown in the Appendix of [19]. In the case of D-MSR, the communication schedules become more sparse and more channels are allocated to the communication schedules, as network density increases. In summary, the spatial reuse of communication resources in D-MSR improves the throughput in the large-scale network.

4.5.5 Reliability in the Network

Several techniques are used in industrial technologies to ensure reliable wireless communication, such as re-transmission, channel hopping, and multipath

routing. The re-transmission scheme depends on the re-transmission of failed packets. In case of errors, this scheme incurs significant communication overhead as well as additional latency in delivering the packets. In multipath routing technique, each node has multiple next hops to forward the packet. When interference causes disruption of communication between a node and its next hop, an alternative path can be used to transport data [16]. Channel hopping and re-transmission schemes are used in the data link layer of D-MSR and WirelessHART. The multipath routing technique is deployed in the WirelessHART standard. In this section, we first evaluate the packet delivery ratio in case of internal interference as well as the effect of re-transmission capability. Next, we assess the behavior of D-MSR and the WirelessHART routing mechanism in terms of reliability in case of extensive external interference. To this end, we set up a number of experiments to evaluate the performance of data delivery in case of lossy networks.

4.5.5.1 Data Delivery Ratio in Case of Internal Interference

In the previous evaluations, we assumed that the interference and transmission ranges are equal and that the two hops reuse distance guarantees interference-free communication in one cell. However, in a realistic setting, the interference and transmission range of a node may not be equal. To address this issue, we evaluate the relation between packet delivery ratio and increased internal interference in the network, in the first experiment. We define five scenarios in D-MSR. In order to assess the worst-case scenario, in the first scenario (D-MSR s1), D-MSR deliberately does not attempt to release the interfered communication link and the MAC re-transmission is not used. In the second, third, and forth scenarios (D-MSR s2, s3, and s4), the re-transmission with one, two, and three retries limit is used in the MAC layer. The fifth scenario (D-MSR s5) combines the advantages of both the re-transmission scheme and the *virtual* link method (discussed in Section 4.4.6.3). For WirelessHART, we have one scenario (WirelessHART s1) that does not use MAC re-transmission. Figure 4.20 shows the data delivery ratio for those six scenarios (D-MSR s1-5, and WirelessHART s1), in case of different interference to transmission range ratios. It is noticeable that the increase in interference range causes more internal interference in the network thereby decreasing the data delivery ration in D-MSR s1-4. On the other hand, D-MSR s5 and WirelessHART s1 provide more reliability in coping with internal interference in case of different interference ranges. This is because D-MSR s5 combines the re-transmission by virtual link method, while WirelessHART s1 avoids the spatial reuse of communication resources.

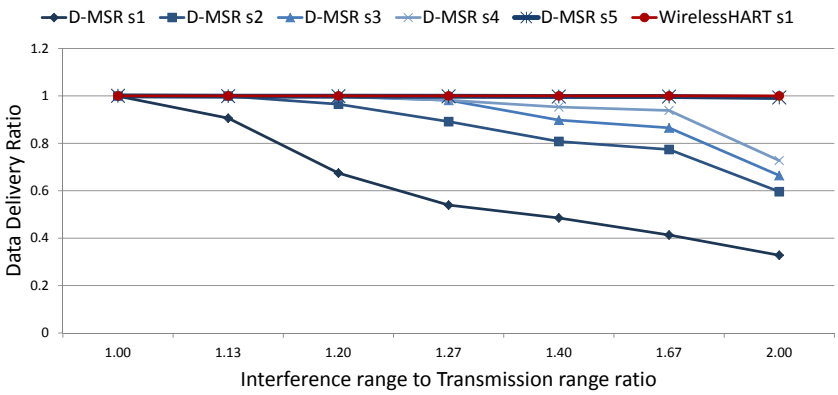


Figure 4.20: Data delivery ratio differences in case of internal interference

In summary, the spatial reuse of communication resources in D-MSR (without considering the re-transmission techniques and virtual link method) is prone to reduced reliability due to internal interference.

4.5.5.2 Data Delivery Ratio in Case of Lossy Network

In this section, we evaluate the behavior of the D-MSR and the WirelessHART routing mechanisms in terms of reliability. In the second experiment, we assume that the edges can only have two states, namely working or failed. We first increase the percentage of broken edges in the network and then measure the number of connections (from the 29 connections that were defined in the previous section) that are still working, i.e., connect the sensors to the actuators. Figure 4.21 shows that in the WirelessHART network, thanks to its multipath routing scheme, more than 50% of the connections are still usable upon increasing the percentage of broken edges to 30%. In contrast, for D-MSR we have around the same, 50% loss of end-to-end connections, when only 10% of edges are broken.

However, thanks to the distributed nature of D-MSR, it can cope faster with interference (or edge failures) than WirelessHART, which uses the centralized approach. There is, therefore, a trade-off in applying the multipath routing in WirelessHART and in applying the distributed scheme to cope with the interference.

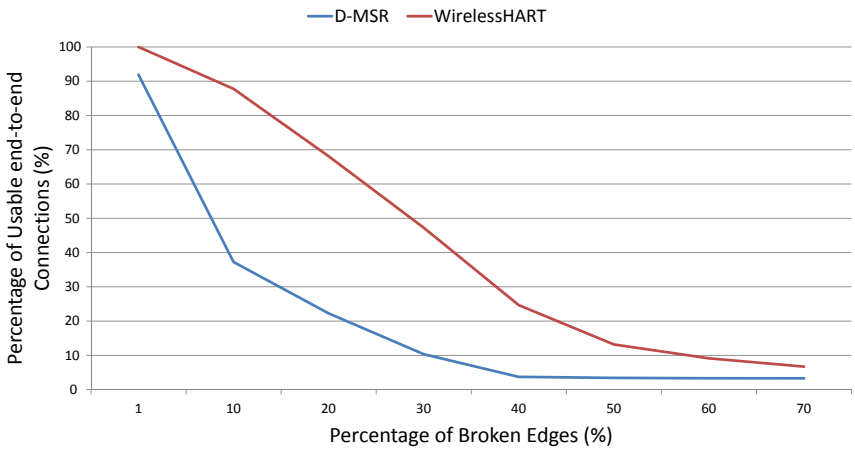


Figure 4.21: Reliability of the end-to-end connections

In the third experiment, we evaluate the relationship between packet delivery ratio and increased interference in the network. To do so, we forward the CBR traffic (periodic sensor data) from sensors towards actuators, for all the 29 connections. At the destinations/actuators we then measure the number of received packets. Unlike in the earlier experiment outlined above, we now assume that the quality of edges may vary from 0% to 100%. In this experiment, we gradually increase the interference regions in the network in six steps (each step takes 2,000 seconds).

These six steps of applying interference in the network are shown in Figure 4.22 and listed in Table 4. In each region, we randomly apply a different interference value to the edges between the nodes. This is because in a realistic harsh environment, each device may experience various packet loss ratios during the communication with each of its neighbors, which may be caused by external interference, non-line of sight connections, multipath fading or the shadowing effect. In this experiment, we assume that the more interference applied to an edge, the higher the chance that the packets will get lost.

We define two scenarios in D-MSR. In the first scenario (D-MSR s1), we gradually increase interference in the network in six steps, while the sensor data are forwarded between the sensor and actuators pairs. Following this, we measure the packet delivery ratio of all the connections between the sensor

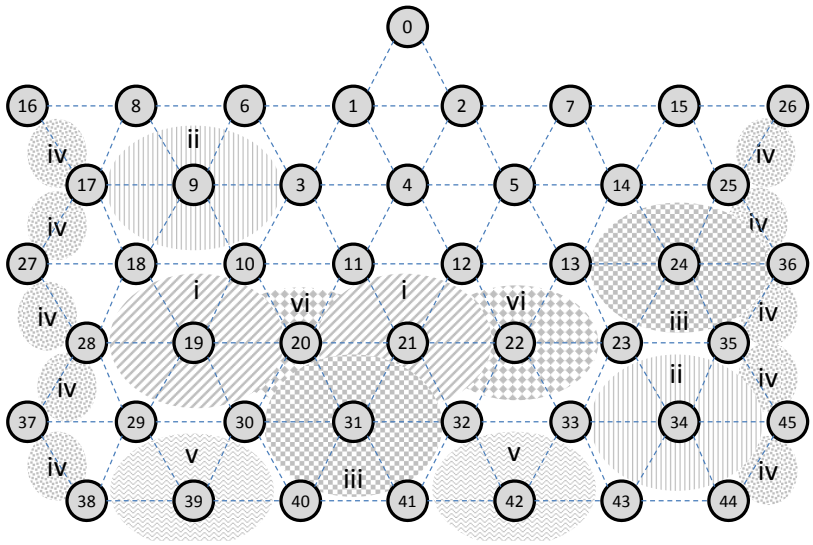


Figure 4.22: Applying interference incrementally in six steps

Table 4.3: Interference steps details

Interference Steps	Affected Edges
Step-i	**19–10, 19–18, 19–20, 19–28, 19–29, 19–30, 21–12, 21–11, 21–20, 21–31, 21–32, 21–2**
Step-ii	**9–6, 9–8, 9–3, 9–10, 9–18, 9–17, 34–23, 34–35, 34–45, 34–44, 34–43, 34–33**
Step-iii	**31–30, 31–32, 31–40, 31–41, 24–14, 24–25, 24–36, 24–35, 24–23, 24–13**
Step-iv	**16–17, 17–27, 27–28, 28–37, 37–38, 26–25, 25–36, 36–35, 35–45, 45–44**
Step-v	**39–38, 39–29, 39–30, 39–40, 42–41, 42–32, 42–33, 42–43**
Step-vi	**22–12, 22–13, 22–23, 22–33, 22–32, 20–11, 20–10, 20–31, 20–30**

and actuator pairs. In this scenario, in order to assess the worst-case scenario, D-MSR deliberately does not attempt to re-construct the routes and re-schedule communication.

In contrast, in the second scenario (D-MSR s2), the routing layer adjusts the affected routes, the D-SAR protocol releases the previous allocated resources on the affected paths and reserves new resources along the new path. Meanwhile, the data are being forwarded through the connections and the packet delivery

ratio of all the connections between the sensor and actuator pairs are being measured.

For WirelessHART we define three scenarios. The first scenario (WH s1) operates under similar conditions as that of D-MSR s1 to also assess the worst-case scenario in WirelessHART. To this end, the WirelessHART network manager does not adjust the affected routes and each node selects its next hop randomly based on the WirelessHART protocol.

In the second scenario (WH s2), we assume that the condition of the poor interfered edge will be reported to the network manager conform the WirelessHART protocol. The network manager re-establishes new graphs through the least affected edges, releases the previously reserved resources on the old path, and then reserves new resources along the new graph/route. These instructions are forwarded towards the network devices upon filling the communication tables of the devices. In this scenario, each node selects the next-hop neighbor on the given graph in a random manner, in line with the WirelessHART protocol.

In the third scenario (WH s3), similarly to the second scenario, we assume that the network manager re-establishes new graphs. However, to pursue better data delivery ratios, each node chooses the best next-hop neighbor based on local information on the packet loss ratio of each neighbor. This mechanism is used in each node during data delivery, while the network manager is collecting information on the edges health status, re-establishing new routes, re-constructing new communication schedule, forwarding the new instruction to the network, and after the maintaining process is finished during normal operation of network.

Those five scenarios (D-MSR s1, D-MSR s2, WH s1, WH s2, and WH s3) are shown in Figure 4.23. D-MSR s2, in which the routes are repaired and resources are re-allocated, performs better than WH s2, in which the network manager re-constructs the interfered graphs and nodes select the next hop in a random manner and even better than WH s3, in which the nodes select their best next-hop neighbors based on their local information (an action which is not listed in the WirelessHART protocol). For instance, after three interference steps have been applied in the network, the data delivery ratio measured is around 7.5% more for D-MSR s2 compared to WH s3 and 41% more compared to WH s2. This large difference between D-MSR s2 and WH s2 can be explained by two facts. Firstly, in WirelessHART more edges are defined in the uplink and downlink graphs to increase their reliability. If the interference in question is extensive in the network, the repaired graphs still may include some poor edges as well. Therefore, if the nodes randomly choose the next hop, these poor edges

may also be selected by them. Secondly, the centralized nature of WirelessHART requires more delay and overhead to fix the problem in the network that greatly affects the data delivery ratio. However, WH s3, in which the nodes select their best next-hop neighbors based on their local information, outperforms the WH s2 regarding packet delivery ratio. Figure 4.23 does not consider the overhead of the repairing phase in terms of delays and the number of required communications.

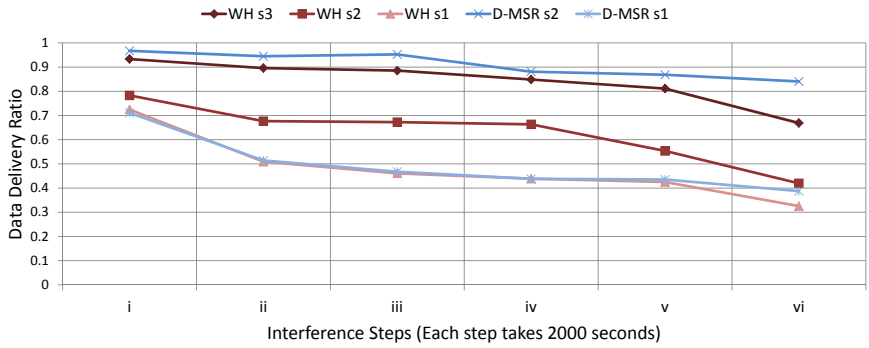


Figure 4.23: Data delivery ratio differences in five scenarios

As is shown in Figure 4.23, the performance of WH s1, in which the network manager does not repair the interfered edges and in which each node selects the next hop randomly, nearly matches that of D-MSR s1, in which the interfered edges and routes are not repaired. The fact that in WirelessHART more edges are defined in the uplink and downlink graphs, does not increase the probability of success in delivering the data in the case of extensive interference and random selection of next hop.

In summary, in the worst case scenario when the two protocols do not attempt to re-construct routes and re-schedule communication, D-MSR s1 performs close to the WirelessHART multipath routing mechanism (in WH s1). However, in the second scenario, the distributed approach (D-MSR s2) assures a higher data delivery ratio than WirelessHART (WH s2 and WH s3). As can be concluded from the first experiment, applying the multipath routing scheme in D-MSR, as a management scheme with a distributed nature, will provide more reliability in data delivery in case of link failures.

In Figure 4.24, we evaluate the repairing mechanism applied in D-MSR

(D-MSR s2) and in WirelessHART (WH s2) to show which mechanism achieves a stable data delivery ratio the fastest. We consider five measuring points in each step to have more detailed view of data delivery ratio changes between interference steps. That way, we can show the behavior of each scheme in recovering the data delivery ratio after applying the interference in each step. As Figure 4.24 shows, the data delivery ratio suddenly drops each time the interference is applied in the network. As expected, D-MSR requires less time to reach the stable data delivery ratio value in comparison with WirelessHART. For instance, after applying the second interference step it takes more than 900 s for WirelessHART and around 500 s for D-MSR to reach a stable state value. This is because D-MSR needs less time to re-construct the new routes, to release the previous resources along the interfered route and to reserve new resources along the new path. In this figure, the AVG value in each step represents the average of the data delivery ratio in that step.

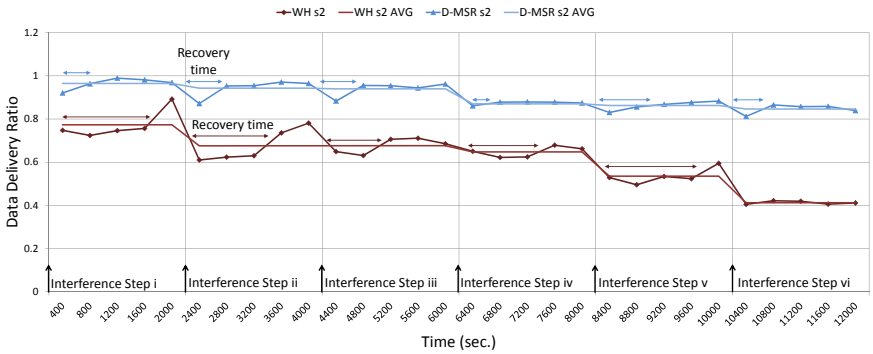


Figure 4.24: Data delivery ratio difference in two scenarios

4.5.6 Power Consumption in the Network

In this section, we evaluate the energy-consumption of network nodes in D-MSR and WirelessHART. The simulation runs for 1,000 s. We measure the total consumed energy at every node during the simulation period. We consider two states of network operation, namely operation in (1) a static and (2) a dynamic environment (e.g., link failures). In the static environment we measure the energy needed to exchange network management messages (periodic updates), as

Table 4.4: Periodic messages rates

Item	Parameter	Value	Transmission type
WirelessHART Periodic management data	Health report rate Advertisement rate	90 s 2 s	Acknowledged unicast Un-Acknowledged broadcast
D-MSR Periodic management data	RPL control message rate Advertisement rate	20 s 2 s	Acknowledged unicast or Broadcast Un-Acknowledged broadcast
Application Data for Both WirelessHART and D-MSR	Sensor data rate	2 s	Acknowledged unicast

Table 4.5: Energy-consumption per transaction and its formula

Notation	Formula	Value
Acknowledged Tx	$TsCCA * Listenpower + TsMaxPacket * Transmitpower + TsAck * Receivepower$	303 μ J
Acknowledged Rx	$TsMaxPacket * Receivepower + TsAck * Transmitpower$	311 μ J
Broadcast Tx	$TsCCA * Listenpower + TsMaxPacket * Transmitpower$	252 μ J
Broadcast Rx	$TsMaxPacket * Receivepower$	264 μ J
Idle Rx	$TsRxWait * Listenpower$	136 μ J

well as application data messages (from sensors to actuators). For the dynamic environment we measure the energy consumed for the network maintenance.

The periodic management messages generated by each device in the WirelessHART network are network health reporting and status commands (i.e., WirelessHART command 779, 780, and 787) and advertisements. In the D-MSR, each device broadcasts the advertisement packets, and the RPL periodically sends controlling messages to maintain routes. Management and application data messages for both D-MSR and WirelessHART are listed in Table 4.4.

Table 4.5 shows the energy-consumption (in this simulation we assumed the energy-consumption in Tx/Rx turnaround, and the processing energy can be neglected) required for each type of transaction. In addition, the idle listening energy at an unused scheduled link is shown. This is the energy that is consumed by the receiver while it is waiting for a message to arrive. Values of the parameters used in Table 4.5 formulas are listed in Table 4.6.

The distribution map of energy consumption for network management traffic and application traffic in the case of static environment is illustrated in

Table 4.6: Energy-consumption parameters

Parameter	Value	Parameter	Value
Radio chip	CC 2420	TsRxWait	2.2 ms
Transmit power (0 dBm)	57.42 mW	TsAck (26 bytes)	0.832 ms
Receive power	62.04 mW	TsCCA	0.128 ms
Listen power	62.04 mW	TsRxTx (TxRx turnaround)	0.192 ms
TsMaxPacket (133 bytes)	4.256 ms		

Figure 4.25. The total energy consumption over the network for management and application traffic is provided in Table 4.7. The total energy consumption for network management is higher in D-MSR than in WirelessHART, which can be explained by the higher data rate of control messages in RPL. From the network management energy map we can see that the distribution pattern for WirelessHART is symmetric, reflecting the regularity of the multi-path routing graph, with bottlenecks being the nodes close to the access points. The distribution pattern in the D-MSR management energy map, is also reflecting the structure of the RPL routing tree, with bottlenecks created at nodes close to the access points.

The application traffic energy map of WirelessHART shows bottlenecks close to the access points, which is due to the fact that all traffic should pass through the gateway. The energy consumption at bottleneck nodes in WirelessHART is higher than in D-MSR bottlenecks. The total energy consumption in WirelessHART is also higher than in D-MSR, which is due to the fact that RPL routing in D-MSR forwards traffic through shorter routes that do not necessarily pass via the gateway. Depending on the position of sources and destinations in the network, the bottlenecks in D-MSR can be more spread in the network area compared to the concentration of bottleneck nodes in WirelessHART. The distribution pattern in the total energy maps is more affected by the application traffic energy pattern.

Figure 4.25 also shows the energy consumption for idle listening. This energy depends on the efficiency of the scheduling mechanism. The better the scheduling, the less energy is needed for idle listening. As a response we did not include that energy in the total consumed energy in the network.

Table 4.7 also lists the consumed energy for network maintenance messages in case of 3–9 link failures. D-MSR requires less overhead and less maintenance energy for coping with disturbances (e.g., link failures) in the network. In Section 4.5.7.3, we evaluate the performance of D-MSR in coping with network

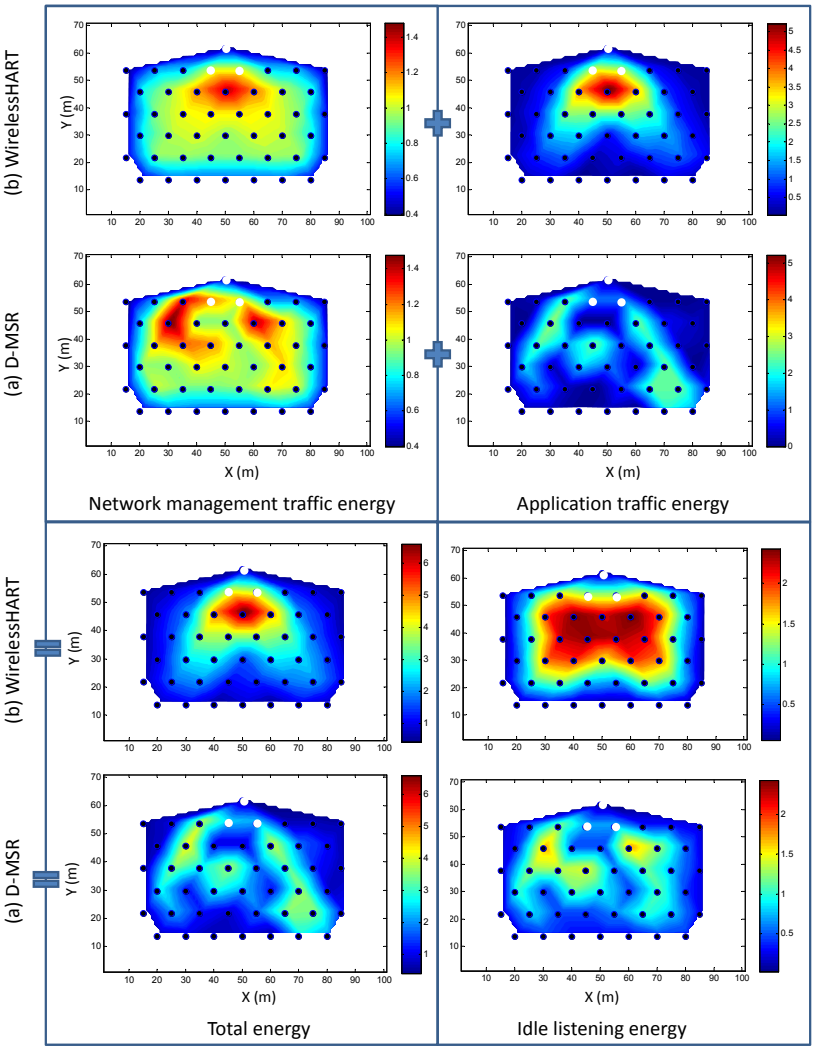


Figure 4.25: The energy consumption distribution in WirelessHART (a) and D-MSR (b)

Table 4.7: Energy-consumption in the network (in 1,000 s) during normal operation.

Environment	Item		WirelessHART	D-MSR
Static	Network management energy		38.77 J	40.70 J
	Application traffic energy		52.84 J	42.71 J
	Total Energy (without idle)		91.61 J	83.41 J
	Idle listening Energy		51.48 J	31.34 J
Dynamic (edge failures)	Network recovery energy	3 edges	0.717 J	0.261 J
		6 edges	1.177 J	0.435 J
		9 edges	1.421 J	0.560 J

dynamicity in more detail.

4.5.7 Evaluating Management Efficiency

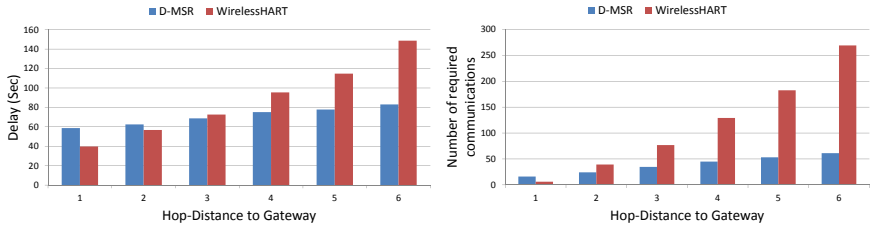
4.5.7.1 Performance During Node Joining

In this section we evaluate the procedure of node joining in D-MSR as well as in WirelessHART, in terms of delay and overhead. We group the nodes based on their distance from the gateway into six categories in our evaluation. In D-MSR, the joining delay for each node at different hop distances in phase 1–3 (i.e., from the moment the node is started up till the node finds its path toward the gateway and the other devices) is the same. This is because most of the communications in phase 1–3 occur locally and do not depend on the hop distance of the node from the gateway. However, in Phase-4 as the hop distance increases, the delay in reserving the management resources increases as well. This is caused by the fact that in Phase-4, each node needs to reserve the management resources along the path towards the gateway and conversely. As the hop distance increases, the reservation procedure takes more time.

To compare our node joining procedure with that of WirelessHART, we consider the total delay and overhead during Phases 1–4 in D-MSR. This is because in WirelessHART, the nodes that have sent the join request to the network manager, must wait to receive the activation command from the network manager after all the necessary network management resources (such as graphs and communication links) have been configured and reserved along the path. The joining procedure in WirelessHART therefore consists of forwarding the join request towards the network manager, allocating the required management resources for all the nodes along the path, and finally forwarding the activation

command towards the new device.

Figure 4.26 displays, the delay in nodes' joining and the number of required communications (number of messages sent) for node joining in the case of different hop distance categories. It is noticeable that the increase in hop distance results in more delay, and in a larger number of communications for joining the nodes. They do so for both D-SAR and WirelessHART.



(a) Node Joining delay (D-MSR vs. WirelessHART)

(b) Node Joining overhead (D-MSR vs. WirelessHART)

Figure 4.26: Nodes joining delay (a) and overhead (b) (D-MSR vs. WirelessHART)

Figure 4.26 indicates that there exist considerable difference in the delay and number of required communications in node's joining between D-MSR and WirelessHART. It shows that the distributed scheme can perform far better in large-scale networks. It implies that D-MSR performs better in those scenarios in which the node joins and leaves the network frequently.

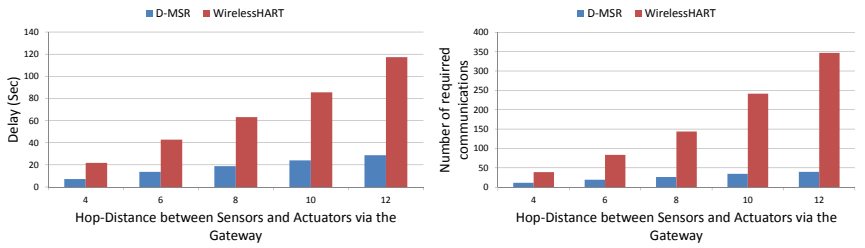
4.5.7.2 End-to-End Connection Establishment between Field Devices

In this section, we evaluate the management efficiency in reserving the communication resources and establishing end-to-end connections between 29 pairs of sensors and actuators. We classified connections into five categories based on the total hop distance of sensor to actuator via the gateway. Figure 4.27 displays, the delay in establishing connections (reserving communication resources) and the number of subsequent required communications for establishing those connections.

It is noticeable that the increase in the total hop distance of the pairs results in more delay, and in a larger number of communications for establishing connection. This is so for both, D-MSR and WirelessHART, but less severe for D-MSR.

Figure 4.27 indicates a considerable difference in the delay and the number

of required communications between D-MSR and WirelessHART. For example, when the total hop distance of sensors to actuators comprises 12 hops, the average of the connection establishment delay is around 75% less for D-MSR compared to WirelessHART, while the average number of required communications for connection establishment is 88% less. Part of this difference can be explained by the fact that in WirelessHART, the network manager has to define more edges to provide a reliable uplink and downlink graph. Subsequently, more communication schedules have to be constructed for those graphs. As a result, more management commands to write the graphs and links are forwarded toward the network devices. The remaining difference could be due to the fact that D-MSR and WirelessHART use different management approaches. Whereas D-MSR relies on the distributed approach, WirelessHART makes use of the centralized management approach, which is far more expensive in terms of time and resources.



(a) Connection establishment delay (D-MSR vs. WirelessHART)

(b) Connection establishment overhead (D-MSR vs. WirelessHART)

Figure 4.27: End-to-end connection establishment delays (a) and overhead (b) (D-MSR vs. WirelessHART)

4.5.7.3 Coping with Changes and Disturbances in the Network

In this part, we evaluate the performance of D-MSR in coping with changes in the network. Figure 4.28 shows the different behavior of D-MSR and WirelessHART in the case of different numbers of edge failures, which are chosen randomly, thereby implying different hop distances from the gateway. We increase the number of edge failures from 1 to 10 and measure the delay, and the number of required communications for coping with edge failure in D-MSR and WirelessHART.

In case of edge failure in D-MSR, the connection manager releases all the re-

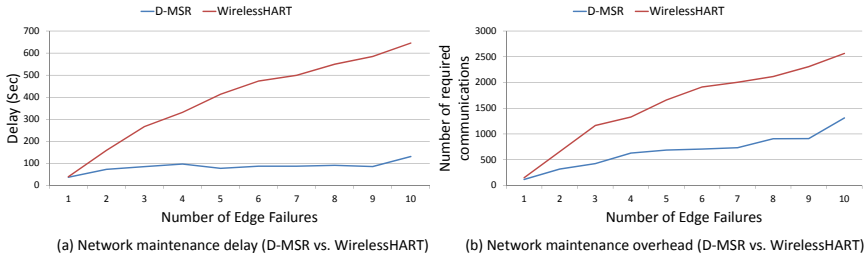


Figure 4.28: Network maintenance delays (a) and overhead (b) (D-MSR vs. WirelessHART)

served communication resources that have failed. The routing layer establishes the new routes then the connection manager re-establishes new connections by reserving resources along the new routes. The delay and overhead in re-establishing connections are shown for both D-MSR and WirelessHART in Figure 4.28.

In WirelessHART, even though the network may still work through an alternative path when graphs are unreliable, the implemented system management algorithm is set to establish new graph and construct new communication schedule. Moreover, it needs to report the edges failure to the centralized network manager, who subsequently establishes new routes, releases the previous schedules on the old routes and construct new communication schedule (links) on the new routes. In D-MSR, this procedure is done in a distributed manner and this causes the relatively low delay and number of required communications. For example, when the number of broken edges is 10, the number of required communications for network maintenance is on average 48% less for D-MSR, compared to WirelessHART. Furthermore, the network maintenance delay is 79% less.

4.6 Conclusions and Future Work

This chapter presented a distributed network management scheme to address the real-time, reliability and throughput requirements of monitoring and process control applications in industrial automation. The resource reservation technique is used in D-MSR for allocating and reserving the communication resources along the path between two end-points (sensors and actuators/gateway).

Channel hopping technique is used in D-MSR to prevent external interference and multipath fading in order to provide a reliable communication. This chapter showed that D-MSR is more efficient than WirelessHART in managing the network when it comes to node joining, reserving the communication resources (either to exchange management packets or sensor data packets), and coping with network dynamicity (e.g., node or edge failures) in terms of latency and overhead. Furthermore, in case of extensive external interference, D-MSR requires less time to reach the stable data delivery ratio value in comparison with WirelessHART.

The spatial reuse of communication resources in D-MSR improves the throughput in the large-scale network at the potential cost of reduced reliability due to internal interference. That is because concurrent transmissions in the same cell may cause transmission failure even when the edges are two hops away from each other, since in a realistic setting the interference and transmission range may not be equal. On the other hand, by avoiding the spatial reuse of communication resources in WirelessHART, the throughput is reduced. This makes WirelessHART less suitable for large scale networks.

Control in the field is not recommended by WirelessHART network. The network manager in WirelessHART supports peer-to-peer sessions between sensors and actuators if the resulting communications are routed via the gateway. This results in more energy consumption by the nodes close to the access points. D-MSR, On the other hand solves that problem by enabling peer-to-peer sessions communication in the field without involving the gateway or access points. This also results in lower energy consumption over the whole network.

The end-to-end delay in D-MSR is close to that of WirelessHART. This result shows that D-MSR can address the real-time requirements, while also achieving a higher efficiency in the network management than WirelessHART, in terms of delay and overhead. Even though the results are promising already, the following points are expected to improve the capabilities of D-MSR.

4.6.1 Supporting Multipath Mechanism in the D-MSR

To provide reliable communication between the endpoints, multi-path routing is used in the routing layer. If a node fails or an edge is broken, an alternative path can be used for delivering the packets. This scheme is applied in several industrial wireless standards, such as WirelessHART and ISA100.11a. We intend to consider this capability in the routing layer. In Chapter 5 we propose multi-path routing and extend the D-SAR signaling protocol to reserve the communication resources along multi-path routes.

4.6.2 Avoiding the Spatial Reuse of the Communication Resources and Improving Reliability

In this chapter, we assumed that the two hops reuse distance guarantees that concurrent transmissions in the same cell will not cause internal interference. In a realistic setting, however, the interference and transmission range of a node may not be equal. This may cause internal interference between those concurrent transmissions. To improve the reliability of D-MSR, we proposed a solution by considering the virtual links that represent the interfering links. We intend to assess another potential solution in which the communication resources (timeslot-channel matrix) are divided into several timeslot-channel blocks and the authority of each block of resources is delegated to different two-hop neighborhoods in a distributed manner. We intend to add this capability to D-MSR to avoid the spatial reuse of communication resources in order to address the requirements of those applications for which improving the reliability is more important than losing high throughput.

In Chapter 6 we propose an extension to ISA100.11a in which the timeslot-channel blocks are delegated to routers by a central system manager based on the routers' requests. The routers manage the star-sub-network, including the I/O devices.

4.6.3 Applying Reactive Discovery for Point-to-Point Routes

Process closed-loop control applications require peer-to-peer sessions between sensors and actuators. In those applications, sensor data periodically streams from sensors to the actuators without needing to involve the gateway or central controller. The RPL used in the D-MSR routing layer, is not recommended to be used for a peer-to-peer traffic mode. That is because, when sensor and actuator need to communicate, the sensor data are restricted to travel in the "up" direction toward a common ancestor and is then forwarded "down" toward the actuator. This scheme may also result in traffic congestion near the gateway. We intend to use a source-initiated reactive extension of the RPL protocol called P2P-RPL [75] in the D-MSR network layer. P2P-RPL enables the field devices to discover the shorter routes to one or more field devices on demand and addresses the point-to-point traffic model requirements without the mentioned drawbacks.

4.6.4 Supporting Point-to-Multipoint in D-MSR

During the resource reservation in D-MSR, we focus on establishing a point-to-point connection between one sensor and one actuator node. In certain industrial closed-loop control applications involving a sensor and multiple actuators, raw sensor readings are streamed from the sensor to the actuators. In traditional Fieldbus technologies such as Foundation Fieldbus, WorldFIP, and ControlNet, certain sensor nodes (the publishers) produce information that they publish to the network. Other groups of sensors or actuators (the subscribers) that are interested in that information listen to the publishers and update their local copy. This scenario can also occur in the wireless approach. In this case we have to consider establishing a point-to-multipoint connection. A point-to-multipoint connection allows one end point to send its traffic to two or more endpoints. The endpoint which generates the traffic is referred to as the root of the connection, whereas an endpoint that receives this traffic is referred to as a leaf. This feature exists in ATM networks and we intend to use the same concepts to add this capability to D-MSR.

CHAPTER 5

D-MHR: A Distributed Management Scheme for Hybrid Networks to Provide Real-time Industrial Wireless Automation

Current wireless technologies for industrial application, such as WirelessHART and ISA100.11a, are not designed to support harvester-powered I/O devices (sensor/actuators), where energy availability varies in a non-deterministic manner. The centralized management approach of these standards makes it difficult and costly for harvester-powered I/O devices to re-join in the network in case of power failure. The communication overhead and delay to cope with the dynamic environment of a large-scale industrial network are also very high for an I/O device. In this chapter, we therefore propose a distributed management scheme named D-MHR, which can address the requirements of energy constrained I/O devices. It is based on the IEEE 802.15.4e and RPL standards. In D-MHR, the routers can dynamically reserve communication resources and manage the I/O devices in the local star sub-networks. We demonstrate that D-MHR achieves higher network management efficiency compared to IS100.11a standard, without compromising the latency and reliability requirements of industrial wireless networks.

5.1 Introduction

Day by day, wired industrial networks are being replaced by wireless solutions. While this creates new opportunities, challenges also arise. For example, the I/O devices in wireless monitoring and process control applications should last for a long time without maintenance. To enable such a working condition, harvester-powered I/O devices with or without additional power sources are increasingly applied. However, state-of-the-art energy harvesters designed for wireless sensor networks, can only generate sufficient power for a limited number of message transmissions/receptions per reporting cycle. Moreover, the availability of the harvested energy often varies in a non-deterministic manner over time. As a result, the harvester-powered I/O devices might frequently loose their connection with the network [76, 14].

In industrial scenarios, three types of network topologies are commonly considered, namely the star, mesh, and hybrid star-mesh topology [77]. In the mesh topology, all nodes (routers and I/O devices) are considered to have routing capabilities. However, harvester-powered I/O devices might not be able to perform routing tasks due to their limited energy budgets. On the other hand, I/O devices can be defined as nodes, with or without routing capabilities, in a hybrid star-mesh topology. This topology is therefore more appropriate for devices with constrained resources and we adopt it for our network.

The network management approach (e.g., centralized management, distributed management) also influences on the suitability of harvester powered devices in the network. Centrally managed networks have limitations in this perspective. First of all, when a harvester powered I/O device has to re-join such a network upon loosing its connectivity, the overhead is too high. The node needs to exchange many messages for this, which incurs high latency. Secondly, in a harsh and dynamic industrial environment, the link between an I/O device and a router may break due to the time varying nature of the channel. To fix such poor/broken links, a central network manager needs to send new instructions over several hops to the network devices, which takes a long time [19]. This problem is further exacerbated as the network scales up. In contrast, a distributed network management approach can address these challenges in a real-time manner with low overhead.

In this chapter, we therefore present a Distributed Management scheme for Hybrid networks to provide Real-time communication (D-MHR) in industrial wireless automation. The key features of D-MHR are as follows:

1. It allocates the communication resources (a set of timeslots) to the routers

in a distributed manner to facilitate real-time communication.

2. The routers in D-MHR are able to manage the I/O devices by forming local sub-networks.

3. The harvester powered I/O devices in D-MHR can choose the best neighbor routers based on their requirements.

4. It constructs the multi-path routes between the routers and reserves the communication resources along the path to provide end-to-end real-time communication.

The remainder of this chapter is organized as follows. The related works and the motivation of this work are discussed in Section 5.2. Section 5.3 describes D-MHR principles. Section 5.4 outlines different management phases of the D-MHR scheme. Section 5.5 compares various performance evaluation matrices of D-MHR with ISA100.11a. Finally, Section 5.6 concludes this work.

5.2 Related works

Several wireless communication standards based on IEEE 802.15.4, such as ZigBee Pro [9], WirelessHART [13] and ISA100.11a [12], are developed to support industrial applications. ZigBee Pro is not designed to support industrial process control applications, which have strict latency and reliability requirements. WirelessHART and ISA100.11a are the two standards most widely accepted by the industry. Both of these standards are managed by central network manager. WirelessHART supports full mesh topologies, while a hybrid star-mesh topology is considered in the ISA100.11a network.

To the best of our knowledge, no industrial wireless standard has been developed by utilizing the distributed management approach thus far. This leads to the creation of the IETF Working Group 6TiSCH to address this issue; their proposed standards are still in a draft state. However, several academic works have focused on this area, which can be divided into two categories: node-based management and cluster-based management. Both node and cluster-based management schemes can also utilize multi-channel communication to improve the scalability and reliability in wireless sensor networks [78].

The node-based multi-channel MAC protocols, such as MMSN [79], MC-LMAC [80], Y-MAC [81], D-MSR [19] and MCMAC [82], try to assign different channels (communication resources) to nodes in a two-hop neighborhood to

avoid potential interferences and to increase network throughput. These protocols, however, face practical issues in real WSNs, including: (a) scheduling overhead and (b) high protocol complexity that may not be suitable for constrained power I/O devices in practice [78]. The cluster-based multi-channel protocols such as TMCP [78] and [83], assign a different static channel to each cluster. These schemes are less complex and more suitable for the constrained power I/O devices. However, these solutions do not consider the advantage of dynamic channel hopping, which is utilized in our work.

5.3 D-MHR: novel concepts and the stack architecture

We propose a cluster-based multi-channel distributed network management scheme (D-MHR) to address the requirements of harvester powered I/O devices. This scheme is based on two standards: IEEE 802.15.4e (TSCH mode) [29] and Routing Protocol for Low power and Lossy Networks (RPL) [15]. We used standards in our work to promote a acceptance in the industry. Simply combining these standards does not work. Instead, their proper integration into a single working scheme is very important, which therefore constitute a key focus area of this chapter. In this work, routers act as cluster-heads.

5.3.1 Overview of D-MHR

D-MHR supports a hybrid network topology as shown in Figure 5.1 (a). The network topology has two levels, the routers form a mesh network, while the I/O devices are part of local star networks. In D-MHR, the RF space is modeled as a matrix of time and channel offset. Time is divided into discrete time slots and a collection of time slots creates a *superframe*. A sample superframe and two cycles of the sample superframe are shown in Figure 5.1 (b) and (c) respectively. A single element in the superframe is called a *cell* and a group of consecutive cells is called a *segment*. A segment may contain, 1, 2, 4 or any factor of the superframe length of cells. A sample of possible segment sizes is shown in Figure 5.2. A particular router (cluster head) can reserve multiple segments to manage its local sub-network and to enable future local communication in that sub-network. As the segmentation size decreases, the resource reservation becomes more dynamic and flexible and can support different traffic characteristics of the network. However, small segmentation size increases the management

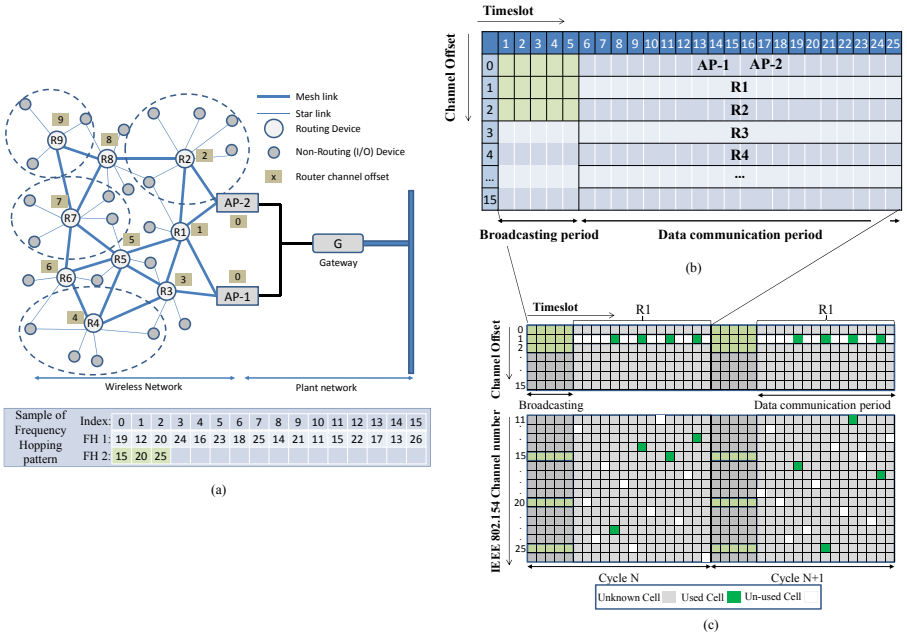


Figure 5.1: (a) D-MHR topology, (b) a sample D-MHR superframe and (c) two cycles of the sample superframe.

overhead to initiate and update the resource reservation in a distributed manner. Selecting the optimal segmentation size for resource reservation, i.e. choose between low complexity and high flexibility and vice versa, is beyond the scope of this chapter. In this chapter, we consider a complete row in the data communication period of the TSCH superframe (i.e., a channel offset) as a segment. Routers (cluster heads) use their chosen segment(s) to manage their local sub-network. All routers divide the communication resources among themselves by selecting different channel offsets in a distributed manner as shown in Figure 5.1 (b) and as further explained in Section 5.4.1.2.

The I/O devices first get synchronized with the system after which they select the best two routers to provide reliable/redundant paths. The I/O devices use the local statistics of the neighboring routers (e.g. RSSI), as well as the advertised global rank (the qualifying numbers defining the router's individual position relative to other routers with respect to the Gateway) of the routers

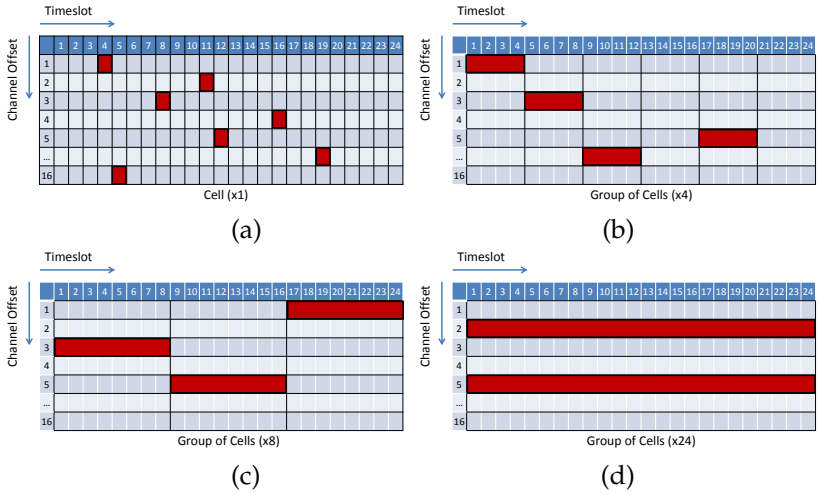


Figure 5.2: A sample of possible segment sizes: (a) $\times 1$ cell (b) $\times 4$ cells (c) $\times 8$ cells, and (d) $\times 24$ cells

to choose the best possible routers according to their requirements. To further communicate with the selected routers, the I/O devices use the communication resources (segments) reserved by the routers. In order to provide real-time communication and to reserve the communication resources toward the final destination, the I/O device informs the routers of its traffic characteristics. This includes specified bandwidth and latency information as well as the communication type (periodic or non-periodic). In this chapter, we assume the prevalence of periodic data traffic between sensors and actuators. The required resources along the multipath routes toward the final destination are reserved by following the D-SAR signaling protocol [22]. D-SAR is a distributed scheduling protocol that is based on concepts derived from ATM networks, which reserves communication resources based on the traffic characteristics requested by the source node.

Due to channel hopping and multichannel communication, the process of joining and neighbor discovery are challenging issues. Another issue is the scheduling of broadcasting links in a distributed manner. To address these, we modified the TSCH matrix by dividing the superframe into two periods: (i) the broadcasting/advertisement period and (ii) the data communication period as

shown in Figure 5.1 (b). The broadcasting period facilitates neighbor discovery. In the broadcasting period, nodes either broadcast their control messages (e.g. advertisements, routing layer messages) or listen to their neighbor's control messages. As no further unicast communications are scheduled in this period, effective data sharing between the nodes is guaranteed. To facilitate faster neighbor discovery and data sharing during a joining phase (especially for harvester powered I/O devices), we limit the number of channels used in that period to three channels namely, 15, 20, and 25. These three channels do not overlap with any of the three common IEEE 802.11 channels and hence less interference occurs in these channels [12]. In the data communication period, the routers choose particular channel offsets to provide unicast communications. The network devices may in turn use the broadcasting and data communication periods to create a superframe of any length that is an even multiple of a basic superframe length (e.g. 250 ms), in which these periods are repeated. A sample superframe of D-MHR is illustrated in Figure 5.1 (c). For example, router R1 selects channel offset 1 by following the respective frequency-hopping pattern illustrated in Figure 5.1 (b). Router R1 uses physical channel 12 in the first slot of the data communication period based on the IEEE 802.15.4e physical channel calculation scheme (FH [1] =12) [29]. Any neighbor of router R1 (either an I/O device or a router) that wishes to transmit to router R1 in the first slot, will set their channel to the receiving channel of router R1 (i.e. 12).

5.3.2 D-MHR protocol stack architecture

The protocol stacks of ISA100.11a and D-MHR are shown in Figure 5.3. In ISA100.11a, a central system manager schedules all the communications and constructs all the routes through the data link layer management object (DLMO). It also establishes end-to-end connections in the network through the transport layer management object (TLMO). In contrast, the network setup is performed in a distributed manner in D-MHR.

The new sub-layers, modules and tables of our proposed D-MHR protocol stack are highlighted in Figure 5.3 (b). The data link layer consists of two sub-layers: the lower and the upper data link sub-layer. In the lower data link sub-layer, we modify the IEEE 802.15.4e (TSCH mode) standard to fit our requirements. A *Two-hop Channel Offset* table is added in this layer enable the allocation of the communication resources to the routers and to enable the scheduling of interference-free communications in the network. In the upper data link sub-layer (the resource reservation layer), we implement D-SAR signaling protocol that is designed to reserve the resources in the multi-path

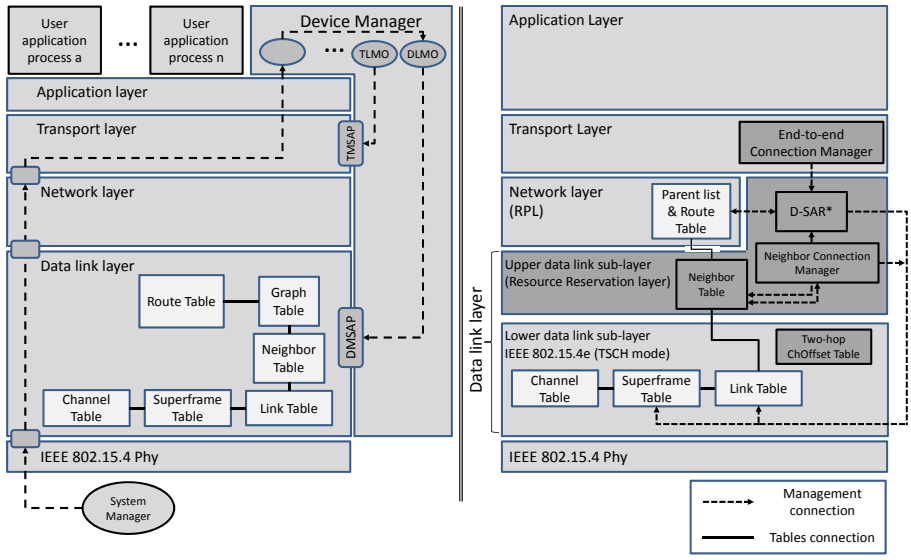


Figure 5.3: The protocol stacks of (a) ISA100.11a and (b) D-MHR.

routers. We also implement the neighbor connection manager modules, that is designed to define the initial communication link between the network devices [19]. This helps to configure the communication tables locally in the lower data link sub-layer. Additionally, the *Neighbor* table containing neighbor statistics, is implemented in this sub-layer. In the network layer, Routing Protocol for Low power and Lossy Networks (RPL) is used with proper adjustments [15]. The *End-to-end Connection Manager* module is implemented in the transport layer. This module establishes the end-to-end connection through the D-SAR signaling protocol.

5.4 D-MHR management functionality

This section describes how a wireless node (either a router or an I/O device) can join the network, discover its neighbors, select suitable routers (i.e. the parent in RPL) and ask for communication resources to enable its further communications. Then, we discuss how routers with management capabilities use their own local

resources to address the I/O devices' requirements by allocating the required bandwidth to them.

5.4.1 Router start-up, joining and maintenance

In D-MHR, it is assumed that routers have enough capabilities and resources to manage a star-network with I/O devices. The routers act as local system managers to their own sub-network. They handle the joining procedure and assign management communication resources by following the steps mentioned below.

5.4.1.1 Joining and neighbor discovery

In this phase, the new router scans the available physical channels and collects the advertisements (or beacons) from the neighboring routers. Then it selects the best advertisers and sends the association (or join) request to them. Upon acceptance, the advertiser transmits a join response/activation command to the new router. The new router follows both the same procedure as explained in [19] and the IEEE 802.15.4e standard to join a network and discover the neighboring routers.

5.4.1.2 Selection of an un-used advertisement cell and channel offset

Upon receiving the activation command from the selected parent router, the new router can start broadcasting its advertisement. To do so, the new router has to choose a free advertisement cell in the broadcasting period. The router also chooses a free channel offset to manage the scheduled communications with its local sub-network and to communicate with other routers in the network.

D-MHR includes some important information in the advertisement of each router, such as (i) advertisement cell numbers, (ii) channel offset numbers of the corresponding router and its immediate neighbors. This effectively allows a receiving router to gather advertisement cells and channel offsets information on its two-hop neighborhood. This enables the routers to choose a free advertisement cell as well as a channel offset in a distributed manner. We assume that the two-hop information guarantees that two routers which are in interference range, do not transmit at the same time, and hence do not cause collisions. As a result, two routers, which are two-hops away from each other, can choose the same advertisement cell or channel offset. If a node selects a timeslot to send the advertisement, the node will transmit an advertisement

in the assigned channel most of the time. If it chooses not to transmit in that timeslot, it listens in a randomly selected channel (after having chosen from three advertisement channels) to receive advertisements from other neighbors. If a node is not scheduled to send the advertisement in a timeslot of the broadcasting period, the node will once again listen in a randomly selected channel.

We also assume that the network density allows the routers to find a free advertisement cell or free channel offset [84]. In case of a dense network, in which there are insufficient communication resources to enable the routers to find a free advertisement cell, we can increase the superframe length. To solve the channels offset issue in the dense network, we can decrease the segmentation sizes as discussed in Section 5.2. In such a case, D-MHR can consider a segment with 4, 8 or any factor of superframe lenght of cells, instead of using a complete row in the TSCH superframe as a segment. As the channel offset is assigned to each node upon joining, there are no longer any concerns over channel offset allocation to the links. This is because the senders set their channel offset to the receiver's channel offset during the communication scheduling. The remaining issue in reserving the communication resources is the allocation of common timeslots among the neighboring routers. Therefore, in the D-SAR signaling protocol, the neighboring nodes (in each hop) negotiate in order to find an unused common cell based on only the information on timeslots, while the channel-offset information is excluded.

5.4.1.3 Initial communication establishment with neighbors

After joining the network, the new router needs to find the route towards other nodes (including the gateway). The neighbor connection manager module of each network device, uses a handshaking mechanism in order to define one Tx and one Rx link with each of its neighboring routers [19]. Those links and a typical management superframe (i.e. 2s) will be added to the data link layer communication tables. These links enable a router to communicate with all its neighbors. After this, the routing protocol can be run to find the path between the endpoints.

5.4.1.4 Route construction

D-MHR uses RPL in the routing layer to find a path towards the gateway. By generating RPL control messages, the routing entries in the intermediate nodes as well as a complete path toward the new router will be constructed. Several control messages are periodically forwarded through the network to maintain

and update the "up" (multipoint-to-point) and "down" (point-to-multipoint) routes. To select the best routers as parents, routers in D-MHR use the following information; (i) the *Neighbor Router* table statistics in the data link layer as local information and (ii) the advertised rank of the neighbor routers based on different objective functions (OFs), included in the RPL control messages, as global information.

Multipath routing in RPL In order to increase redundancy/reliability and load balancing in the network, it is desirable to use a multipath route between a source and the final destination. In RPL, we assume that all the routers store the routing information. Upon receiving the sensor data or management messages from the previous child, each router chooses the next hop randomly from the two best parents in the "up" direction. This enables reliable multipath routing in RPL in the "up" direction. To enable multipath routing in the "down" direction, the prospective destination node (router or I/O device) sends/forwards Destination Advertisement Object (DAO) messages to its two best parents and finally to the gateway. As a result, the routing table in the intermediate routers, stores the potential multipath routes in the "down" direction.

5.4.1.5 Contract or end-to-end connection establishment

In D-MHR, the D-SAR signaling protocol is used to reserve resources in a distributed manner to exchange management packets toward the Destination Oriented Directed Acyclic Graph (DODAG) root (i.e. gateway) [22].

Resource reservation scheme in multipath routing In the D-SAR signaling protocol, the source node sends the *setup* message toward the destination node along the route defined by the routing layer. The setup message includes parameters such as a list of suggested common unused timeslots for further communication with the next hop, a final destination address, traffic ID, and a requested publishing period. The receiver of the setup message then performs a check of its available communication resources. If the required resources are available, the receiver chooses one timeslot from the suggested free timeslots, based on the requested publishing period of the traffic. The selected time slot is then allocated by writing a new link and (if needed, new) superframe in the related tables of the data link layer. In the next step, the receiver (intermediate node) forwards the setup message toward the destination node. This process continues until the destination node receives the setup message. The destination

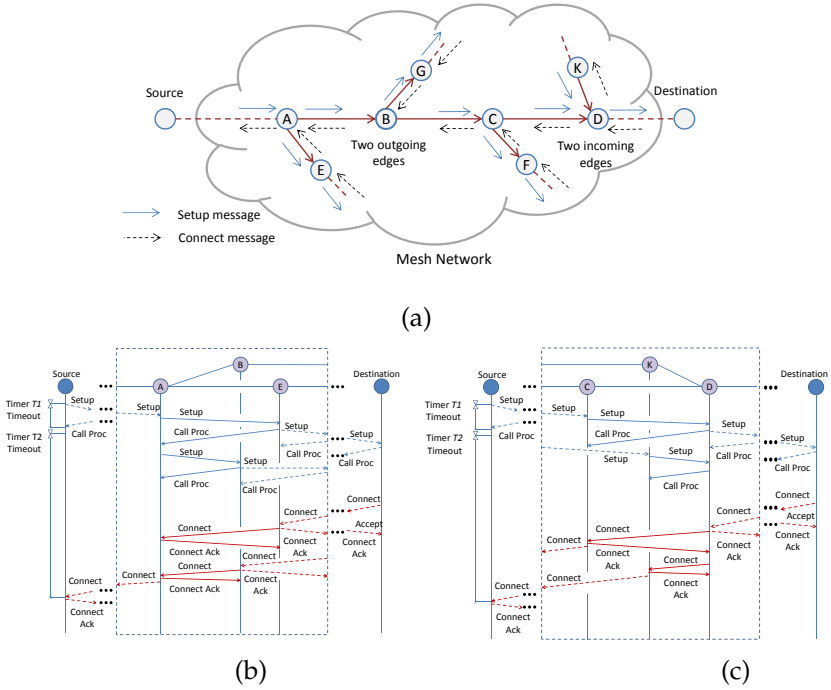


Figure 5.4: Overview of connection establishment protocol in D-SAR extension protocol.

node can either accept or decline the new connection request from the source node by sending the *connect* message or *release complete* message. This connect message traverses along the multi-hop network back to the source node. All the temporary communication resources, which are reserved during the setup message exchange, are switched to permanent reservation.

An alternative path from multipath routing can be used in case of node failure or a broken edge. We modified the D-SAR protocol to be able to reserve the communication resources in the potential reliable multipath routes. In the D-SAR extension protocol, each node sends the setup message to both potential next-hop neighbors as shown in Figure 5.4 (a). Node A, which has received the setup message, forwards it to both of its neighbors (node B and E) in the route. In this case, every node with two outgoing edges in each branch, should also receive two potential connect messages. For example, node A, B or C might

receive two connect messages. Upon receiving both potential connect messages, the node forwards one connect message toward the source node, as shown in Figure 5.4 (b).

If a node (e.g. node D) in a branch with two incoming edges, receives two setup messages, it no longer forwards the second setup message. Upon receiving the connect messages, which is the reply of the first setup message, it sends two connect messages to both setup senders, toward the source node. Should a second setup message be received after the first connect message has been sent, the connect message will be sent immediately to the second setup message sender. In the D-SAR extension, the nodes in the branches with two outgoing edges induce the responsibility of the source node to wait and collect all the connect messages, as shown in Figure 5.4 (c).

To reserve the resources in the "up" direction in the multipath RPL, each node (either source or intermediate node) sends the setup message to its two highest ranked potential parents. This process continues until the setup messages reach the DODAG root. As a response, the D-SAR signaling protocol extension is able to reserve the resources in the "up" direction in the multipath RPL. To reserve the resources in the "down" direction in the multipath RPL, each node sends the setup message to both potential next-hops in the routing table. These two potential next-hops are added to the routing table, when the two potential DAO messages from a final destination is received from these two next-hop children. The D-SAR extension signaling protocol waits in the branches with two outgoing edges to receive the two potential connect messages, and then sends one connect message to the parent.

The I/O device is typically a multi-hop away from the final destination (either actuator or gateway). Due to the earlier discussed mesh routing, there might be multi-path routes toward the final destination. By allocating the required communication resources in each hop, based on the sensor traffic characteristic, most of the reserved resources might be wasted. This is because only one path among several alternatives is selected to forward the traffic, during the normal operation of the network. As a response, when a device or router has two successors acting as next hops, the transmission rate in the setup message will be reduced to half of the original sample rate. A similar approach is used in the centralized scheme in [39]. All the I/O devices or routers follow this policy to reduce the transmission rate in the signaling protocol in the intermediate branches. Eventually, by accumulating all the reserved resources on each edge of the multi-path routes, the requirements of the original transmission rate of the source nodes will be fulfilled.

5.4.1.6 Coping with internal interference in the network

In D-MHR, two routers that are two-hops away from each other can reuse a channel offset. In a realistic setting, the interference range of a node may be much larger than its transmission range. When two pairs using the same selected channel offset, communicate concurrently, interference will be unavoidable. Thanks to the scheduled communication concepts, this internal interference can be detected by observing the constant packet loss in those cells after reservation. The router that detects the potential conflict can change its chosen channel offset and, subsequently, the I/O devices' channel offsets.

5.4.2 I/O device start-up, joining and maintenance

The steps that an I/O device follows to join the network and start publishing/subscribing the periodic sensor data are explained below. The I/O devices might or might not be powered by energy harvesters.

5.4.2.1 Joining and router discovery

When starting up, an I/O device scans the channels to receive potential advertisements from the neighbor routers. Upon receiving advertisements, the I/O device adds the desired information (e.g. received RSSI, RSQI, RPL rank and router channel offset) to the *Candidate Router* table. The Candidate Router table in D-MHR is similar to the *Candidate Neighbor* table of overheard routers in the ISA100.11a standard. In addition to the Candidate Router table, each I/O device stores the statistics on linked/associated routers in a related table. These local statistics and the information on the routers' rank help the I/O device to choose the best possible router.

Using the statistics stored in the candidate router table, the I/O device selects two best ranked routers for further communication. Then, the I/O device sends join requests to this selected router(s) through the advertised Rx link and listens for advertisements on the Tx link to receive the activation command. The router, upon receiving the join request from the I/O device, will process the join request locally. Following this, the selected router should send an activation command to the I/O device. These tasks resemble the system manager's responsibilities in the ISA100.11a standard.

5.4.2.2 Selection of an un-used advertisement cell

Upon receiving the activation command, the new I/O device starts to send advertisements with less transmission rate compared to routers. The selection procedure of a free advertisement cell is similar to that of a router, as explained in Section 5.4.1.2. However, unlike the routers, the I/O devices do not need to select an un-used channel offset. Any I/O device that is scheduled to transmit to / receive from a router, sets its channel to the router's channel at the scheduled timeslot.

5.4.2.3 Initial communication establishment with the selected routers

In this phase, the I/O devices follow the same procedure as the routers, as explained in Section 5.4.1.3.

5.4.2.4 Router selection and route establishment

The new I/O device chooses the best router(s)/parent(s) based on the following information: (i) the candidate router table statistics as local information and (ii) the routers' rank in RPL in terms of different OFs. To provide reliable routing, each I/O device chooses the two best routers as its RPL parents. During the normal network operation, the I/O device might need to change the routers to cope with possible changes in the network. In that case, the I/O will still use the earlier mentioned information to select the best two routers.

The I/O device, upon selecting the RPL parents and joining the RPL, starts to send the DAO message to its potential parents/routers to construct the "down" path in the network. The routers that have received the DAO, update the routing information in their table. Unlike the RPL routers, the I/O devices (i.e. the RPL leaf) do not advertise the RPL by broadcasting the DODAG information object (DIO) message.

5.4.2.5 Contract or end-to-end connection establishment

The I/O device sends setup messages to both selected routers to communicate with the potential destination (gateway or actuator). These messages also include its traffic characteristics information. To provide real-time communication, it is important to reserve communication resources before the I/O device starts to publish its sensor data. The router forwards the same setup message to the requested destination along the established multi-path routes by RPL,

upon receiving a setup message from an I/O device. The router(s) receives the final connect message from the final destination upon allocating the required resource in the mesh network. The details of reserving the communication resources in multi-path mesh routers are described in Section 5.4.1.5.

The I/O device might decide to leave the router (or might be forced to do so) and choose a new one for various reasons. For example, due to a power failure from an energy harvester. In that case, the router should determine whether it considers the device as being removed or not. The timeout mechanism can be used to decide. Based on the timeout, the I/O device may terminate its contract by sending a *release* message before leaving the router. The router forwards the release message along the multipath routes toward the final destination to free up the allocated resources in the network. The details of releasing the resources are specified in Chapter 4 and in [22].

5.4.2.6 Sensor data publication/subscription

The I/O device, as a sensor node, publishes its sensor data toward an actuator or gateway. The I/O device uses the constructed routers in RPL and the selected parent(s) to deliver the data toward the final destination.

5.5 Performance evaluation

To compare the performance of D-MHR with ISA100.11a, different matrices, such as channel re-use factors, end-to-end packet delivery latency, management efficiency, etc. are evaluated in this chapter. After explaining the simulation setup, we explain the performance matrices below.

5.5.1 Simulation setup

Both the D-MHR and ISA100.11a protocol stacks are implemented in NS-2. We consider a network of 38 I/O devices, 22 routers, 2 access points and 1 gateway in a $80m \times 40m$ area. The routers are placed systematically in the network, while the I/O devices are randomly distributed. The transmission range of all the nodes are considered 15m. We use the two-ray ground radio model in the simulation. The constant bit rate (CBR) traffic model is employed to generate the sensor data in our simulation. The management and application data publishing period in ISA100.11a and D-MHR are listed in Table 5.1.

Table 5.1: Management and sensor data publishing period in ISA100.11a and D-MHR.

Item	Parameter	Value
Management data	Channel and neighbor diagnostics report (ISA100.11a) and RPL control message rate (D-MHR)	30 sec
	Advertisement rate (ISA100.11a and D-MHR)	4 sec
Application data	Sensor data rate (ISA100.11a and D-MHR)	4 sec

5.5.2 Communication schedules and network throughput

In D-MHR, the communications are scheduled by the routers in a distributed manner, as explained in Section 5.4.1.2. Routers far away from each other (more than two-hops) can choose the same channel offset. This means that a same cell can be reused in several neighborhoods. On the other hand, in ISA100.11a, the central system manager schedules all the communication and there is no scope of re-using the dedicated cells in the network. We compare the communication schedules of ISA100.11a (Figure 5.5 (a)) and D-MHR (Figure 5.5 (b) and (c)) for the same traffic scenarios, where the cell re-use numbers are displayed with different colors. As expected, the communication matrix of ISA100.11a uses a particular cell only once.

For D-MHR, we show two different scenarios. In the first case, a router chooses the first available channel offset among the free channel offsets, which are not used in the two-hop neighborhood (Figure 5.5 (b)). Here, some channel offsets are unused, while some cells are reused multiple times in different parts of the network. As shown in Figure 5.5 (b), certain cells, e.g. cells in channel offset 1, are reused by six pair of nodes in different neighborhoods. In the second case, a router randomly selects the channel offset from the free channel offsets (as shown in Figure 5.5 (c)). As a result, the communication schedules are more spread than in the previous case and almost similar to the ISA100.11a matrix, but with cell reuse possibility.

The spatial reuse of communication resources (i.e. channel offsets) in D-MHR leaves 81% and 77% of cells un-used in the first and second schemes, respectively, whereas in ISA it is 64%. The spatial reuse of communication resources in D-MHR helps to improve the network throughput in a large scale-network.

The first scheme of D-MHR can be used to mitigate *external interference*. This is because, it can easily blacklist the problematic channels, either locally

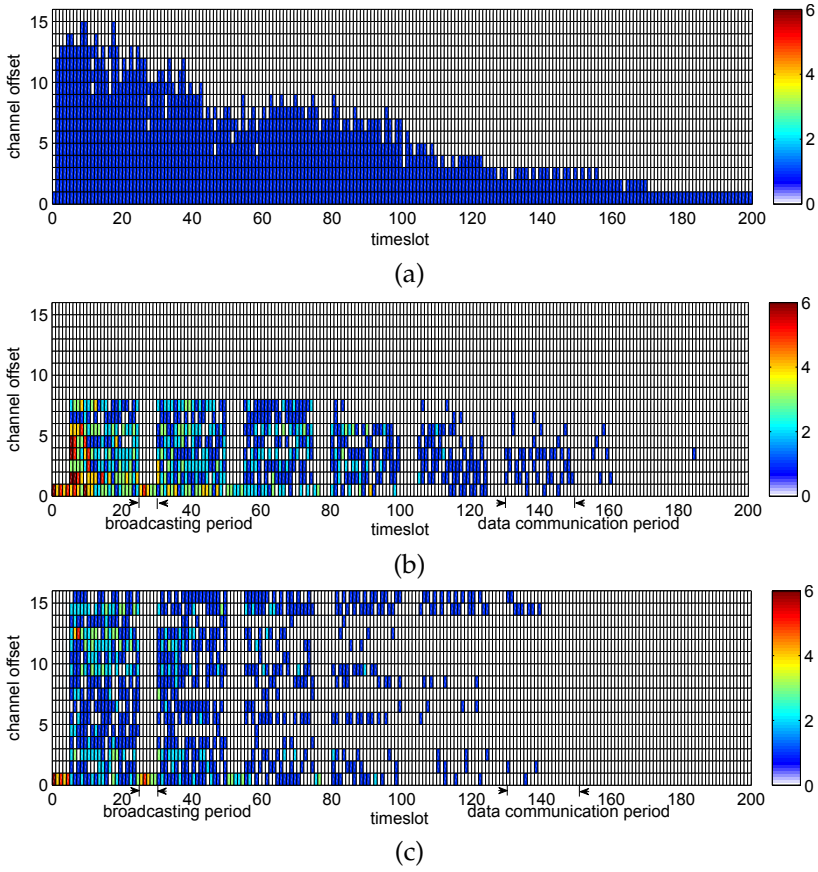


Figure 5.5: Communication scheduling matrices of (a) ISA 100.11a, (b) D-MHR (first available channel offset selection), (c) D-MHR (random channel offset selection).

or in the entire network. The hopping pattern sequence can also be adapted without interrupting the network or without having to re-schedule all the communications. In addition, by deploying more than one antenna or by increasing the number of access points, we can use the un-used channel offsets and increase the network throughput. However, the network might be more vulnerable to *internal interference*. Since in a realistic setting the interference and

transmission ranges may not be equal, the following problem can be occurred. Different pairs of nodes, which are using the same cell to communicate, may cause transmission failure, even when they are two hops away from each other. In such scenarios, the second scheme of D-MHR can provide more robust communications.

To address this issue, we evaluate the relation between packet delivery ratio and increased internal interference in the network. For example, in case of internal interference, the data delivery ratio in D-MHR is 94% and 98% for the first and second schemes, when the interference range is 70% higher than the transmission range. However, the routers can detect the potential conflict and change their channel offset by applying the monitoring scheme (discussed in Section 5.4.1.6). Then the data delivery ratio rises to 100%. On the other hand, in ISA100.11a, in which no spatial reuse of communication resources is assumed, the data delivery ratio is 100%.

5.5.3 Reliability and real-time guarantee

To evaluate the reliability and real-time guarantee of D-MHR and ISA100.11a in the presence of external interference, we introduce failures between I/O devices and routers in the star sub-network. After this, the packet delivery ratio and the time interval of the consecutive packets are calculated at the destination. Figure 5.6 (a) illustrates that the packet delivery ratio suddenly drops for both approaches when we apply the external interference. However, compared to D-MHR, it takes longer for the ISA100.11a standard to reach back to the stable state. Figure 5.6 (b) shows the jitter in the time interval of the consecutive packets received at the final destination. It varies slightly around the expected value of four seconds (data publishing interval) in normal operations. When the interference is applied, the jitter in ISA100.11a dramatically increases and requires considerably more time to reach back to the normal values than in D-MHR. In ISA100.11a, the system manager has to perform repairs on receiving the periodic neighbor diagnostic reports, which takes time. On the other hand, in D-MHR, the I/O devices can use their local statistics to fix the problem, which improves the reliability and real-time aspects of our approach.

5.5.4 Data delivery latency

To evaluate the end-to-end data delivery latency, several end-to-end connections are considered in the network. We also evaluate the potential delay jitter, and the average number of hops that the received packets need to travel to reach

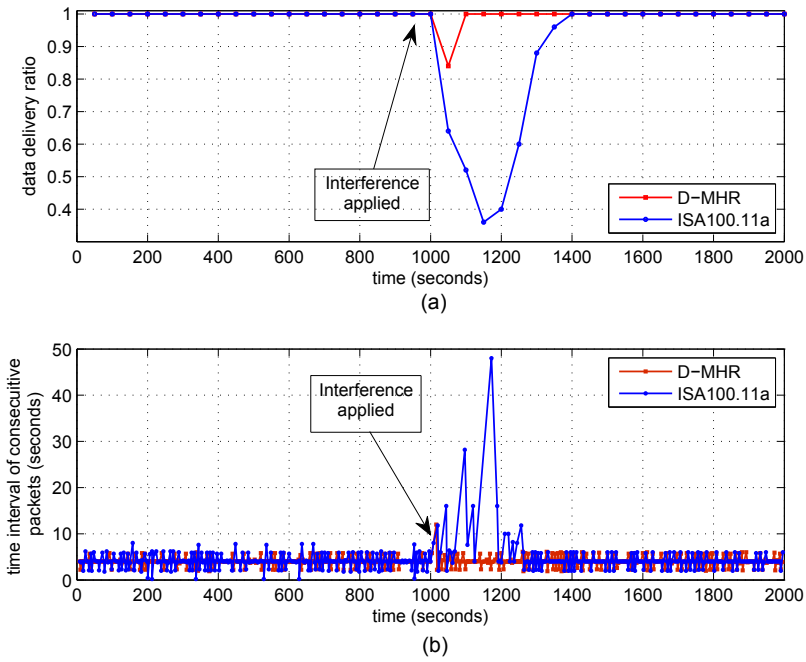


Figure 5.6: Reliability and real-time guarantee.

their destination through the defined end-to-end connections between I/O devices. We classified connections into four categories based on the shortest hop distance between a sensor and its final destination (e.g. the actuator) via the gateway. Beforehand, the required resources are reserved by applying the various mechanisms discussed in each protocol, based on the sensors' traffic characteristics.

In Figure 5.7, we can see that the end-to-end delay in D-MHR is less than in ISA100.11a. The average number of hops that the sensor data travel in D-MHR is less than in ISA100.11a, as is shown in Figure 5.7 (b). This confirms (i) the lower end-to-end delay in Figure 5.7 (a) in the related classification and (ii) the fact that in D-MSHR the packets travel less distance to reach the destination. This difference can be explained by the fact that the data packets in D-MHR (thanks to the usage of RPL in the network layer) may be able to reach the

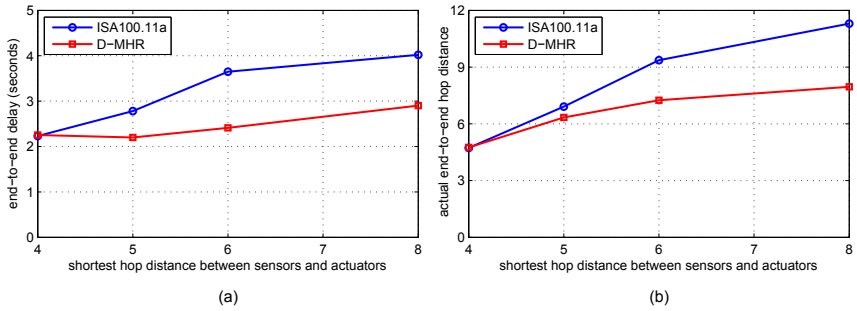


Figure 5.7: (a) Average end-to-end delay and (b) actual hop distance of ISA100.11a vs. D-MHR.

final destination (i.e. actuator) without passing through the access-points and gateway. It is noticeable that in practice, the average number of hops that the packets travel is higher than the end-to-end shortest path shown in the horizontal axis of the figure.

5.5.5 Evaluating Management Efficiency

5.5.5.1 Performance during node joining

To evaluate the I/O joining delay and communication overhead in ISA100.11a and D-MHR, we group the I/O devices based on their distance from the gateway. In D-MHR, I/O devices send join requests to the selected routers and then reserve communication resources for management message exchange between the routers in the mesh network. In the evaluation, we neglect the scanning delay during the joining process for both schemes. The total joining delay and the communication overhead to reserve the management resources are considered. In ISA100.11a, the I/O join requests are forwarded toward the system manager, after which the system manager defines the graph and reserves the communication resources for the new I/O device.

Figure 5.8 (a) and (d) display the I/O's joining delay and the communication overhead (number of messages sent) respectively, for different distances to the gateway in ISA100.11a and D-MHR. It is noticeable that both the joining delay and communication overhead increase significantly in the ISA network with the increase in hop distances. In contrast, in the D-MHR network, the joining delay

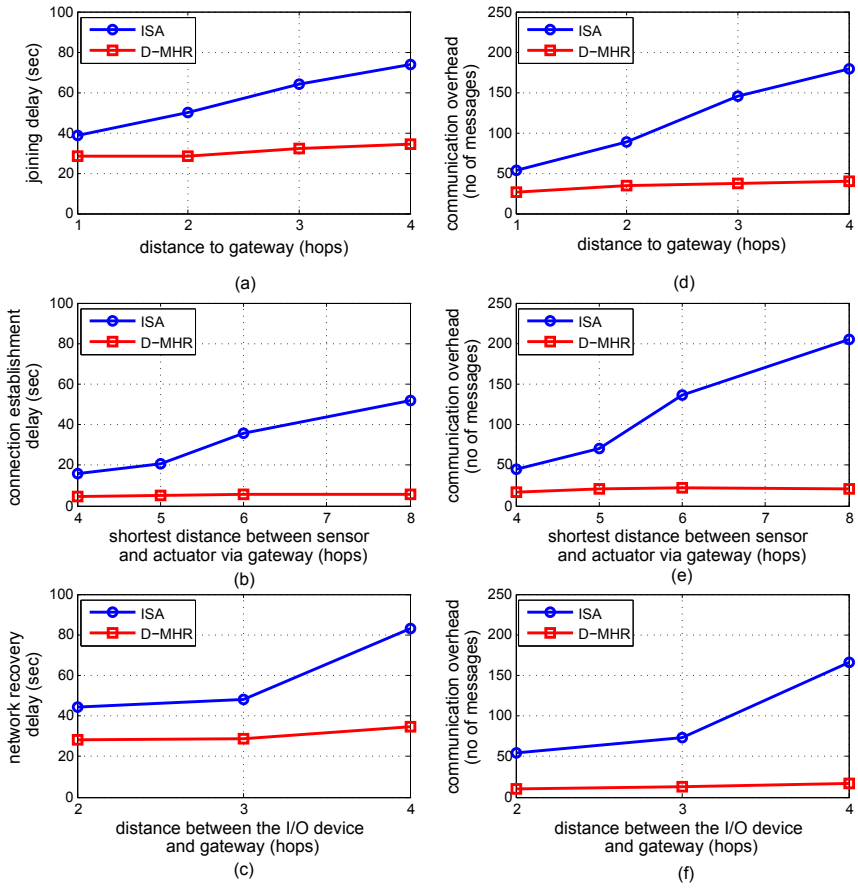


Figure 5.8: Management efficiency comparison between ISA100.11 and D-MHR in terms of *delay* during (a) I/O joining, (b) end-to-end connection establishment, (c) network maintenance, and *communication overhead* during (d) I/O joining, (e) end-to-end connection establishment, (f) network maintenance.

and communication overhead seem to be independent from the corresponding I/O's distance to gateway. Thus, the proposed D-MHR scheme can perform far better in a large-scale network. This also makes them suitable for scenarios in which the harvester-powered I/O devices have to join and leave the network

frequently.

5.5.5.2 End-to-end connection establishment between I/O devices

As explained in the previous section, we grouped the I/O devices based on the hop distances. However, in this section, the distance is calculated from an input device (sensor) to an output device (actuator) via the gateway. This is because we calculate the delay in establishing end-to-end connections by reserving the communication resources between sensors and final destination (i.e. gateway or actuators). Figure 5.8 (b) and (e) show the connection establishment (reserving communication resources) delay and the number of required communications to establish those connections. The connection establishment delay and communication overhead increase along with the hop distances in the ISA100.11a network. On the other hand, in the D-MHR network, the I/O devices can establish the connection much faster than in ISA100.11a and with low message exchange overhead. The delay and communication overhead do not increase if the network scales up. This difference can be explained by the fact that D-MHR and ISA100.11a use different management approaches. Whereas D-MHR relies on the distributed approach (D-SAR signaling protocol), ISA100.11a makes use of the centralized management approach, which is far more expensive in terms of time and resources.

5.5.5.3 Coping with changes and disturbances in the network

In case of network dynamicity, such as edge failure between I/O device and routers due to interference, the I/O devices might have to re-join the network or find a new router. The ISA100.11a standard considers a mechanism in which the I/O device can store allocated resources in its memory. When the I/O device rejoins, it can use the earlier allocated resources to communicate with the routers. This solution can only work in a small-scale network with static situations. However, in a large-scale dynamic network, the network connectivity might frequently change; using the old resources to communicate with the assigned routers might not be useful anymore. On the other hand, D-MHR select new routers based on the rank of the neighbor routers and its local statistics. The I/O in question, sends new joining requests to the new routers and reserves the required communication resources along the path toward the final destination through the new selected routers.

In this section, we evaluate how well D-MHR and ISA can cope with changes in the network. We intentionally introduce interference in the network, which

causes edge failures in different regions of the network. Figure 5.8 (c) and (f) show different behaviors of D-MHR and ISA in the case of edge failures between the I/O devices and chosen routers.

In the ISA network, the system manager chooses two routers for each node (I/O device or router) to increase reliability. Should the I/O device loose one of those routers, the node might send the *connectivity-alert* to the system manager. The system manager may configure new routers instead of the older one. Then the new routes and potential resources might be reserved in the new path. In the event of a connection loss in the D-MHR network, a node may choose a new router based on its requirements. It sends a joining request to the newly selected router and use the D-SAR signaling protocol to reserve the communication resources on the multipath route towards the gateway. This enables reliable and real-time communication with the rest of the network. Due to the distributed manner of the procedure, relatively low delays and low message exchanges overhead are required to fix the edge problem. For example, when the edge failure takes place on an I/O device which is four hops away from the gateway, the communication overhead for the network maintenance is on average 90% less for D-MHR than for ISA. Furthermore, for a central manager it takes a long time to fix a problem in a mesh network. As a consequence, the network recovery delay is 42% higher in ISA than in D-MHR.

5.5.6 Power consumption

To evaluate the energy-consumption of network nodes in ISA100.11a and D-MHR, we consider two states of network operation, namely a static and a dynamic environment (e.g. link failures). In the static environment, we measure the energy needed to exchange network management messages (periodic updates), as well as application data messages (from sensors to actuators). In the dynamic environment, we measure the energy consumed for network maintenance. We run the simulation for 1,000 seconds while the calculations follow the equations and parameters given in [19].

The energy consumption of the ISA100.11a and D-MHR networks in different environments are presented in Table5.2. We utilize the parameters mentioned in Table 5.1. The routers consume on average five times more energy than the I/O devices in both approaches. The network management energy consumption in D-MHR is significantly lower than in ISA100.11a due to the D-MHR data sharing mechanism in the broadcasting period. Unlike ISA100.11a, in D-MHR the nodes are not scheduled in the specific broadcasting links to exchange their data. As a result, they save more energy during broadcasts to their

Table 5.2: Energy consumption in the network.

Environment	Item		ISA100.11a	D-MHR
Static	Average router energy		2.01 J	1.25 J
	Average I/O device energy		0.35 J	0.29 J
	Network management energy		31.32 J	17.18 J
	Application data energy		28.37 J	22.72 J
	Total energy (without idle)		59.69 J	39.90 J
	Idle listening Energy		60.49 J	45.92 J
Dynamic (one edge failure)	Network recovery energy	2 hop distance	0.033 J	0.006 J
		3 hop distance	0.044 J	0.008 J
		4 hop distance	0.105 J	0.01 J

neighbors. The application data energy consumption in D-MHR is also lower than in ISA100.11a, because the RPL forwards the traffic through shorter routes that do not necessarily pass via the gateway. As a result, the total energy in D-MHR is also less than in ISA100.11a. Table 5.2 also lists the consumed energy for network recovery in case of edge failures. D-MHR consumes considerably less energy in the whole network to cope with the edge / node failures than ISA100.11a, due to the distributed management scheme of D-MHR.

5.6 Conclusions and future works

This chapter presented a distributed network management scheme for hybrid networks named D-MHRs, which can support industrial applications by providing reliable and real-time communication. D-MHR can achieve a lower latency in data delivery than ISA100.11a. The nodes can (re-)join the D-MHR network significantly faster than the ISA network with much lower communication overhead. The connection establishment phase is also faster and cheaper in our proposed scheme. Moreover, D-MHR can fix the network problem more quickly and with less message exchanges overhead in case of internal and external interference than the ISA100.11a standard. Thus, D-MSR can better support the monitoring and process control applications in industrial automation, including energy constrained I/O devices (e.g., harvester powered). To further evaluate the performance of D-MHR and ISA100.11a, future works will focus on test-bed implementation.

CHAPTER 6

ISA100.11a*: The ISA100.11a extension for supporting energy-harvested I/O devices

Wireless standards developed for industrial applications such as ISA100.11a and WirelessHART, generally use centralized management approaches. However, such centralized approaches cannot cope with network dynamicity in real-time manner. They also incur high management overhead and latency. Consequently, the network becomes unsuitable for resource constraint devices, e.g I/O devices. The problems become exacerbated when the network scales up. ISA100.11a standard allows reduced functionality devices in the network and supports hybrid network topology. We propose an extension to ISA100.11a to better address the requirements of the energy constrained I/O devices. The proposed extension makes the management more decentralized by delegating a part of the management responsibility to the routers in the network. It also allows the I/O devices to choose their best routers according to the metric considered using local statistics and advertised routers' ranks. We show that the proposed extension can better address the real-time and reliability requirements of industrial wireless networks. It can achieve higher network management efficiency in terms of reducing the delay and overhead of I/O devices than the ISA100.11a standard.

6.1 Introduction

Wireless standards developed for condition monitoring and process control applications have increasingly gained the confidence of industry and their adoption has increased over the last few years. Most of these applications expect the wireless sensor/actuators (I/O devices) to work for long durations of time without maintenance. To facilitate such working conditions, energy-harvested I/O devices with or without additional power sources are becoming popular. The availability of harvested energy typically varies over time in a non-deterministic manner. With today's energy harvesters, only a few wireless transmission/receptions per reporting cycle of the I/O devices are feasible [14]. This calls for the design of efficient wireless communication protocols suitable for industrial environments.

ISA100.11a [12] and WirelessHART [13] are two of the most important standards accepted by the industry. In wireless networks, typical network topologies are either star networks, mesh networks or hybrid networks (a combination of star and mesh). In WirelessHART, all field devices are considered to have routing capability to support full mesh topology. On the other hand, the I/O devices in the ISA100.11a network can be defined as nodes with or without routing capability. It thus supports both star, mesh and hybrid topology. As the harvester-powered I/O devices have severe constraints on resources, especially energy, it is advisable to make them non-routing (end devices) in the network. Hence, the hybrid network topology supported by ISA100.11a is more suitable for them.

The ISA100.11a standard (and also WirelessHART) uses a centralized management approach, which cannot cope with network dynamicity in a real-time manner. The link quality between I/O devices and routers may vary considerably due to the interferences in harsh industrial environments. Rejoining the network and coping with such dynamic situations are costly for I/O devices, as several message exchanges are required to fix the broken links, which incurs high latency [19]. Additionally, the energy-harvested I/O devices might temporarily lose their power as well as their network connectivity, causing additional rejoining processes. These problems are further exacerbated as the network scales up and the I/O devices are several hops away from the central System Manager (SM).

Proper enhancements of the ISA100.11a standard are essential to make it suitable for energy constrained I/O devices. To address this, we propose ISA100.11a*, the extended ISA100.11a standard with a hybrid network management scheme. It makes the management more decentralized by delegating

some parts of the management responsibilities and the authority of communication resources from the central SM to the routers. The routers can schedule communications and address the requirements of the I/O device locally in the star sub-network. The communication schedules and graphs between the routers in the mesh network are constructed by the SM, the same way as in ISA100.11a. Therefore, this hybrid network management scheme proposes a centralized management scheme for the mesh network and a distributed localized management scheme for the star networks.

Another proposed enhancement is the possibility for I/O devices to choose their best possible routers rather than having the SM set these for them. This gives them the flexibility to choose routers and switch easily and quickly to better ones when available. This will improve their efficiency and save the harvested energy.

The rest of the chapter is organized as follows: Section 6.2 summarizes related works. The a brief overview of the concept of ISA100.11a* is explained in Section 6.3. Section 6.4 provides details on the functional description of ISA100.11a* and Section 6.5 evaluates the performance of the proposed approach. Finally, Section 6.6 concludes the work and summarizes our future research goals.

6.2 Related works

A survey on wireless sensor network protocols developed for addressing real-time and reliability requirements in industrial process and monitoring automation is given in [16].

ZigBee Pro [9], WirelessHART, ISA100.11a and IEEE 802.15.4e [85] are the IEEE 802.15.4 [18] based standards. ZigBee Pro, as one of the first standards for WSNs, is designed for applications which have soft real-time and reliability requirements. Since ZigBee Pro runs on a CSMA-based MAC protocol, it is unsuitable for applications that require reliable and timely packet delivery. ZigBee Pro uses *frequency agility*, which is not as tolerant as WirelessHART and ISA100.11a mechanisms to fluctuating wireless conditions and introduces inconvenient delays [16]. It does not support multi-channel communication and hence cannot increase the network throughput.

WirelessHART and ISA100.11a standards are designed for process control and monitoring applications. Both standards support several industrial applications classes with different Quality of Service (QoS) requirements, from monitoring to control [8].

Recent academic studies on time slotted multichannel protocols can be divided into two categories: node-based management and cluster-based management. Both node and cluster-based management schemes can utilize multi-channel communication to improve the scalability and reliability in wireless sensor networks [78]. The node-based multi-channel MAC protocols such as, MMSN [79], MC-LMAC [80], Y-MAC [81], D-MSR [19] and MCMAC [82], try to assign different channels (communication resources) to nodes in a two-hop neighborhood to avoid potential interferences and to increase network throughput. These protocols, however, face practical issues in real WSNs, including: (a) scheduling overhead and (b) high protocol complexity that may not be suitable for constrained power I/O devices in practice [78]. The cluster-based multi-channel protocols such as TMCP [78] and [83], assign a different static channel to each cluster. These schemes are less complex and more suitable for the constrained power I/O devices. However, these solutions do not consider the advantage of dynamic channel hopping, which is utilized in our work.

6.3 Overview of ISA100.11a*

The ISA100.11a standard has several limitations when it comes to supporting resource constrained I/O devices and large-scale networks. A management scheme that speeds up the re-joining procedure of the I/O devices and reduces the overhead of fixing broken links in the network is needed. ISA100.11a* lets the I/O devices (a) (re-)join the network more efficiently by adopting the hybrid network management approach and (b) select and change their parent(s) more efficiently based on changes in the environment.

In the hybrid management scheme, the authority over parts of the communication resources is delegated to the routers to handle the local requirements of the I/O devices in the star sub-network. Based on the number of estimated I/O devices and their local statistics, the routers ask for resources from the SM. Routers use these local resources to allocate management resources to potential I/O devices upon receiving their join requests. The remaining network resources are managed by the central SM, which constructs the routing graphs and communication schedules between the routers in the multi-path mesh topology.

A sample network topology in ISA100.11a* with routers having management capabilities and the corresponding superframe structure are shown in Figure 6.1. The SM manages the first block of resources and uses these resources to define the communication links between the routers in the mesh topology.

The remaining resources are allocated to different routers for their own local management. The size of the blocks allocated to routers is based on expected network load, which can vary according to the number of I/O devices associated with each router.

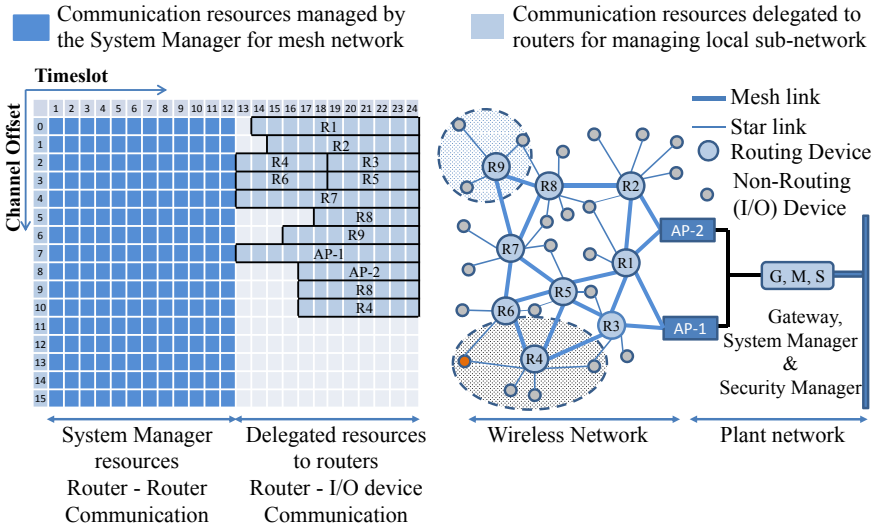


Figure 6.1: A sample network in ISA100.11a* and the superframe structure

The routers use their own resources to send both the join reply and the contract reply in response to I/O devices' requests, unlike the traditional ISA100.11a or WirelessHART networks where they are handled by the SM. The router defines several Tx and Rx links to communicate with the I/O device. The I/O device sends the contract request, including its traffic characteristics to the routers. The router uses its local resources to define more potential links in order to let the I/O device publish its sensor data. The router then forwards a new contract request to the SM to reserve the required resource in the mesh network, based on the I/O traffic characteristic. This speeds up the joining procedure of the I/O devices.

As the energy-harvested I/O devices might frequently shut down and lose their connectivity with the routers, the router should not release the communication resources reserved for the I/O device if no updates about its presence are received. Since the I/O devices are not participating in routing tasks, it is

not necessary to remove them very fast from the network. This policy lets the energy-harvested I/O device work more efficiently in the network.

The next key contribution of the ISA100.11a* is that the I/O devices are able to choose/change the associated routers based on their metrics (e.g. end-to-end latency, reliability, and power consumption). The I/O device keeps the statistics of the overheard neighbor routers in a *Candidate Router* table in which it updates the status of its connectivity with the routers. To let the I/O device choose the best router, it needs to know the ranks of the neighboring routers, which are basically qualifying numbers defining the router's relative position/grade with respect to the Gateway. The routers advertise their rank based on different *Objective Functions* (OFs) (e.g. reliability, latency, power consumption and available bandwidth). This advertising is inspired by the Routing Protocol for Low power and Lossy Networks (RPL) [15]. However, while the routers' ranks are calculated in a distributed manner in RPL, they are calculated by the SM in ISA100.11a*. The SM calculates those ranks based on (1) routing information, (2) schedule information, and (3) the diagnostics/statistics reports that are received periodically from the mesh network and sends them to the routers for advertising. The I/O devices use their local statistics such as RSSI and RSQI and the routers' rank to select the best routers. This will improve their efficiency and save the harvested energy.

In the ISA100.11a standard, the I/O device can store the allocated resources in its memory. When it loses the network connectivity and wants to rejoin, it can use the earlier allocated resources to communicate with the routers. However, in large-scale dynamic networks, the network connectivity might change frequently, and using the old resources to communicate with the assigned routers might not be useful any more. The capability of the I/O devices to choose/change the associated routers in the ISA100.11a* helps faster rejoin in such cases.

ISA100.11a*'s main contributions and extensions can be listed as follows:

- Proposing hybrid network management - managing the mesh network between routers in a centralized manner and managing the star sub-network in a distributed manner.
- Allocating communication resources to routers to address the requirements of I/O devices.
- Calculating routers' ranks based on different OFs by the SM and advertising the ranks by the routers to let I/O devices choose the best routers based on their requirements.

- Letting I/O devices join the network much faster, and re-select their routers according to the metrics considered based on the local statistics and routers rank.

6.4 Functional description

This section describes how a wireless node (either a routing device or an I/O device) can discover its neighbors, join the network, find its router and ask for communication resources for management and data delivery in ISA100.11a*. It also proposes how the routers with management capabilities use their own local resources to address the requirements of I/O devices and allocate the requested bandwidth for them.

6.4.1 Routers' management phases

The routers can be classified as routing devices with and without management capabilities. A router without management capabilities is just the same as a router in ISA100.11a and hence we do not explain their working, but rather focus on the routers with management capabilities. These management capabilities do not increase the complexity of the routers as they run a simple network management algorithm to manage a small star sub-network. The different management phases that guide the routers from startup to the moment they start publishing (or subscribing) their periodic sensor data (as a field device with routing capabilities) or providing management services to the I/O devices (as a router with management capabilities) in their sub-network are discussed below.

6.4.1.1 Startup, neigbour selection and joining the network

Similar to ISA100.11a network, upon start up a router scans the channels, collects the statistics about the neighboring routers in a candidate list and sends a join request via the proxy routers to the SM. It receives the join reply from SM through the potential neighboring routers. The SM selects the potential neighboring routers, based on the received reports from the network and the candidate list included in the router's joining request. After completing the joining procedure, the router has the routes and communication resources for sending management messages such as periodic reports about neighbors and channels statistics to be sent to the SM.

6.4.1.2 Constructing the routes

As in the ISA100.11a, the routers use graph routing to send their data to the final destination. The SM constructs routing graphs and updates them whenever new routers join or network topology changes, based on the *Neighbor Diagnostics* reports or *connectivity alerts* received from the network. The new constructed routes are sent to the routers and written in their routing/graph tables.

6.4.1.3 Contract or end-to-end connection establishment

The routing devices (with and without management capabilities) send contract requests with their traffic characteristics to the SM to reserve the required communication resources in the network for exchanging either application traffic, management traffic or sensor data. The routers with management capabilities ask the SM to reserve the communication resources for their local star-network in addition to the potential initial resources along the multi-path route toward the Gateway in the uplink and downlink direction.

6.4.1.4 Delegating the authority over a block of resources to routers

The authority over parts of the communication resources will be delegated to routers to manage the one-hop star sub-network. The delegation takes place after a negotiation procedure between the router and the SM. The allocated resources (e.g. channel offset or several numbers of cells) are used to address the local requirements of the sub-network as shown in Figure 6.1. Each router is capable of running a simple network management algorithm to manage the small star topology. To provide real-time communication between an I/O device and its destination (the Gateway or an actuator), the routers might also reserve the communication resources beforehand, along the path to the destination in the mesh network.

The communication resources delegated to the routers depend either on the request of the router based on the number of estimated I/O devices in its candidate I/O device table or on a predefined fixed number of cells. The routers might ask for more resources later on, to fulfill their local requirements upon detecting more I/O devices or running out of communication resources due to receiving unexpected joining requests.

Each router updates its neighboring unlinked I/O device statistics and information in the candidate I/O devices table in which the overheard neighbor's address, device type, and statistics are stored. The router uses the informa-

tion about the I/O devices in its candidate I/O device table to reserve some resources for its potential communication with those same I/O devices. The reservation is undertaken either in the local star-network or between the routers in the multi-path routes toward the Gateway. Routers ask the SM to provide resources based on the number of estimated I/O devices and their RSSI and RSQI values.

When an I/O device chooses its router, the router could use the already reserved resources to create local links with the I/O device. Each router will keep the collected statistic information with its linked/associated I/O devices in an I/O Device Neighbor table (similar to Neighbour Diagnostic table in ISA100.11a standard) with several parameters such as Mean RSSI, Packets Received number, and Missed ACK Packet number.

6.4.2 I/O devices' management phases

An I/O device that joins the network through its desired routers might not notice whether the routers are using the distributed or centralized approach. The different management phases that guide an I/O device from starting up to the moment the node starts publishing (or subscribing) periodic sensor data in the network are discussed below:

6.4.2.1 Startup, router selection and joining

The I/O devices start scanning the channels and receive advertisements from the neighboring routers. They collect the overheard neighboring routers' statistics and fill out/update the required information (e.g. received RSSI and router's ranks) in a Candidate Routers table. A list of stored information about the overheard un-linked routers is shown in Table 6.1. The updating rate depends on the capabilities of the device. Each I/O device also maintains a Neighbor Routers Diagnostic table (similar to the Neighbor Diagnostic table in the ISA100.11a standard) to store information about its each linked/associated router (more than one router for reliability) as shown in Table 6.2.

The routers broadcast their ranks in terms of different metrics in the network such as reliability, latency, and power consumption to reach the Gateway. The I/O devices choose the best router(s) based on the routers' rank according to the OF considered and on the local statistics stored in tables. For example, for addressing the reliability requirement, the I/O device uses local information (included in the Candidate Router table or Neighbor Routers Diagnostic table) and the router reliability rank to choose the best one.

Table 6.1: Candidate Routers table

Field name
N(count of discovered routers)
$Router_1$ (16-bit address of first candidate)
Uplink $Router_1$'s Ranks based on different OFs (new)
Downlink $Router_1$'s Ranks based on different OFs (new)
$RSSI_1$ (radio signal strength of first candidate)
$RSQI_1$(radio signal quality of first candidate)
etc ...
$Router_N$(16-bit address of Nth candidate)
Uplink $Router_N$'s Ranks based on different OFs (new)
Downlink $Router_N$'s Ranks based on different OFs (new)
$RSSI_N$(radio signal strength of Nth candidate)
$RSQI_N$(radio signal quality of Nth candidate)

Table 6.2: Neighbor Routers Diagnostic table

Field name	**Description**
Uplink $Router_1$'s Ranks	Received the $Router_1$'s uplink Ranks to GW based on different OFs (new)
Downlink $Router_1$'s Ranks	Received the $Router_1$'s downlink Ranks from GW based on different OFs (new)
$RSSI_1$ (level)	Received signal strength indicator from this neighboring $Router_1$
$RSQI_1$ (level)	Received signal quality indicator from this neighboring $Router_1$
$RxDPDU_1$ (count)	Number of valid Packets received from this neighboring $Router_1$
$TxSuccessful_1$ (count)	Count of successful unicast transmissions to the $Router_1$
$TxFailed_1$ (count)	Number of unicast transmission, without getting any ACK or NACK
$TxCCABackoff_1$ (count)	Number of unicast transmission aborted due to CCA
$TxNACK_1$ (count)	Number of NACKs received
$ClockSigma_1$ (level)	A rough estimate of standard deviation of clock corrections
... (Other routers)	

Upon choosing the best router(s), the I/O device sends a join request to the selected router(s), through the advertised Rx link and listens on the advertised Tx link to receive the join reply. The router processes the request locally, unlike in

the traditional ISA100.11a standard where it acts as a proxy router and forwards the request to the SM. The selected router sends an activation command to the I/O device and writes local resources in the I/O device communication table (e.g. superframes, links, graphs, and channel tables). The I/O device may select more than one router to provide more reliability. In such cases, it sends a new joining request to the second router. The provisioning procedure and the reception of the new network key are not needed in the second trial. However, the I/O device will receive some management resources, including primary links, superframes and graphs to communicate with the second router.

The I/O device then starts to report per-channel and per-neighbor (i.e. Channel Diagnostics and Neighbor Diagnostics reports) to the selected routers. The routers process the received reports locally unlike the traditional way of sending the report directly toward the SM. The routers inform the SM about the I/O devices they support. As a result, the SM and the Gateway know how to reach to the I/O devices through the selected routers.

6.4.2.2 Contract or end-to-end connection establishment

The I/O device sends separate *contract requests* to each selected router, including traffic characteristics information for communication with the potential destination (Gateway or actuator). Before publishing the sensor data, the I/O device needs to reserve the resources (1) between itself and the neighboring routers as well as (2) between the routers in the multi-path routes in the network to the final destination. This resource reservation ensures real-time communication between I/O devices and the destinations.

Based on the communication service type, different schemes might be used to forward the traffic in ISA100.11a*. In case of periodic/scheduled service, resources might be reserved in the slotted hopping period, while in case of non-periodic/unscheduled service the slow hopping and CSMA scheme can be used. In this chapter we consider only the periodic case and assume that the data traffic between sensors and actuators has a constant bit rate. Hence the resource reservation is undertaken in the slotted hopping period.

The router(s) might employ different types of policies when it receives a contract request from the I/O device. It can forward the same contract request to SM as in the traditional ISA100.11a standard. Alternatively, the router hides the I/O device from the rest of the network and sends its own contract request. There, the I/O device acts as a new sensor attached to the router and behaves as a user application process in the router. We consider the first policy in our work, where the routers send a new contract request to the SM to reserve the

communication resources between the routers in the mesh network.

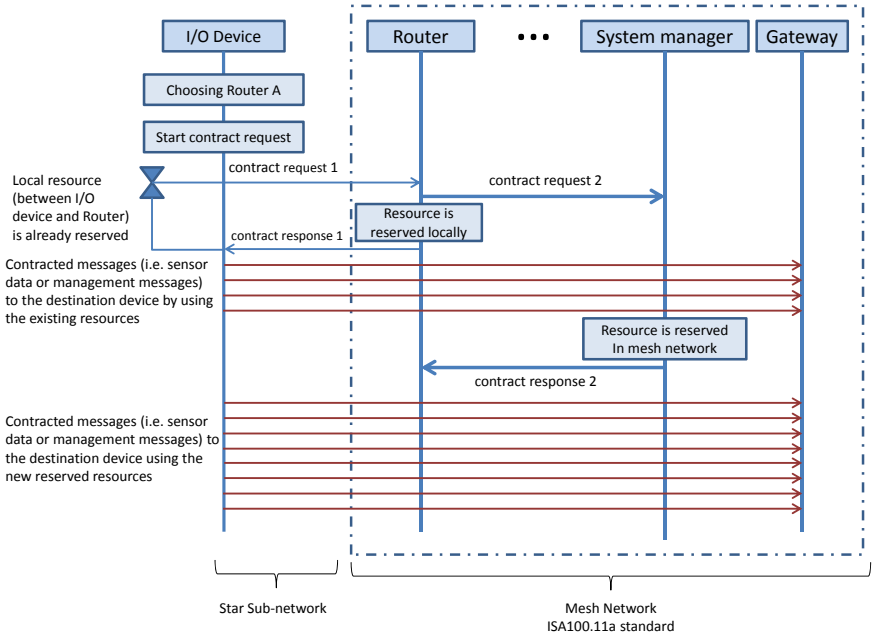


Figure 6.2: Contract establishment

Figure 6.2 shows a sample of a contract establishment mechanism between the I/O device and the router and between the router and the SM. The router that received the contract request from the I/O device allocates resources based on the traffic characteristics for further communication with the device and replies to the device with the contract response. The router uses its own resources, which are already delegated, to write the new links and superframes in the Link table and Superframes table of the I/O device. This allows the I/O devices to start publishing the data faster than the traditional approach as shown in Figure 6.2. If the router's delegated communication resources are not sufficient to address the requirement of the I/O device, the contract response to the I/O device is postponed until the router receives the local (delegated) communication resources and the contract response from SM.

Based on the ATM networks' [36] concepts, the routers can setup the virtual paths to the destination by over-provisioning some resources on the paths.

In such cases, the routers ask the SM to reserve more resources on the path toward the Gateway in the mesh network. As a result, the processing and the contract establishment times for newly joining I/O devices can be reduced. When subsequent virtual channels have the same source (i.e. the selected router) and destination (i.e. the Gateway), they need not be provisioned every time when a new contract request is received from a new I/O device. To optimize over provisioned resources, an efficient estimation of the needed resources is required. This can be done based on the number of estimated I/O devices and their local statistics in each router. If enough resources are not reserved in the mesh network, the router might send a new contract request to the SM to reserve some resources along the multi-path to the final destination, based on the new I/O device traffic characteristics and some additional resources based on the over-provisioning policy. The router receives the final contract reply from the SM upon allocating the required resource in the mesh network. The I/O device receives the contract response from the router much earlier when compared to the traditional approach. Upon receiving the contract response it starts publishing its data to the router and the router forwards the traffic toward the destination by using the existing resources.

6.4.2.3 Contract termination, deactivation and reactivation

The connection quality between the I/O device and the selected routers varies or the neighboring routers' rank might change. As a result, the I/O device might decide to change its selected router and choose a new one. The I/O device terminates its contract by sending a terminate request before leaving the router. Upon receiving the terminate request, the routers release the resources from the I/O device; but based on the over-provisioning policy, they might not free up the reserved resources in the mesh network. Hence the routers, based on their estimation on the number of neighboring I/O devices and their statistics, might send a new terminate request to the SM and might ask for the resources along the multi-path routes to the destination to be released.

When the router determines that an I/O device is no longer part of the network, it shall terminate the contracts associated with that I/O device and free up the network resources that were allocated for supporting those contracts. If the energy-harvested I/O devices lose their connectivity with the network for a while, the router could decide whether it considers the node as being removed or not. A timeout mechanism can be used for this. If the timer expires before receiving any message from the device, the I/O device is considered as removed. The router might release its local resources from the I/O device, or

keep these reserved resources as long as the router still has sufficient resources. When it comes to freeing up network resources that were allocated to the I/O device in the mesh network, different policies can be adopted. Firstly, based on a timeout mechanism or receiving the termination request from the I/O devices, the router might terminate the I/O device's contract with the final destination and free up the network resources. Secondly, the router based on the over-provisioning policy can keep the network resources unless the router's estimation for required resources results in releasing some of the resources in the mesh network or the network runs out of resources leading to the termination of the contract by the SM. The second policy reduces processing and contract establishment times for the future joining I/O devices. In Figure 6.3, a sample of a contract termination mechanism between the I/O device and the router and also the termination mechanism between the router and the SM are shown. These samples assume that the I/O device generates the contract termination request.

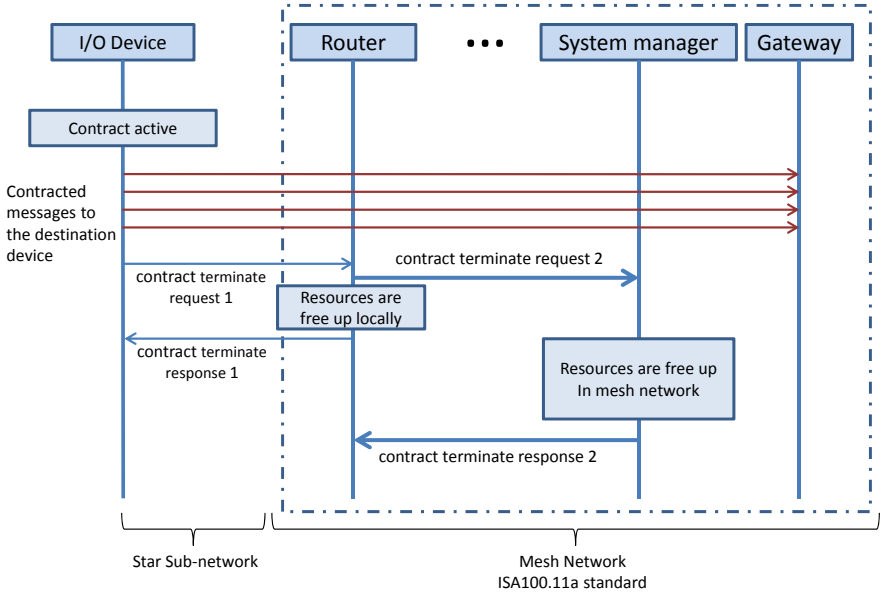


Figure 6.3: Contract termination

6.4.2.4 Publishing (or subscribing to) the sensor data

The I/O device, as a sensor node, publishes its data toward its destination. The I/O device first sends its data toward the assigned router(s), including the destination information. The router uses mesh routing (i.e. graph routing) and forwards the data toward the Gateway or final destination. If the route toward the final destination does not exist, the data will be forwarded toward the Gateway. Unlike the traditional ISA100.11a where SM constructs an uplink (or downlink) graph from the I/O device to the Gateway, here the I/O device uses its selected routers' uplink (or downlink) graph. The selected router acts as a proxy router to reach the Gateway. If the Gateway is not the final destination, it forwards the data toward the final destination. When the direct graph/route toward the final destination might not be available at the Gateway, it uses the selected router(s) of the final destination as a proxy router to reach the destination.

6.4.2.5 Coping with external interference in the network

Similar to the ISA100.11a standard, the I/O device considers adaptive channel hopping on a link-by-link basis [2] in addition to the traditional blacklisting on the whole network. Each I/O device updates the channel and neighboring router statistics in the Channel Diagnostic and Neighbor Routers Diagnostic tables respectively. The statistics include local statistics as well as the rank of the routers. In case of interference in the network, different edges may experience different packet losses and ranks might change. The I/O devices choose the best available routers based on new local and global network statistics. This approach can better cope with disturbance in a large-scale network in a real-time manner compared to the existing ISA100.11a approach.

6.4.3 System Manager Extensions

The SM manages the communication schedules between the routers in the mesh network in a centralized manner and delegates the authority over a block of resources to the routers so that they can manage the star-sub networks locally. The SM constructs the uplink/downlink graph from/to routers to/from the Gateway and schedules the communication in the constructed graph. It also receives the neighboring statistic reports of routers. Hence it has all required information to calculate the global ranking of the routers in the mesh network.

The SM calculate the routers ranks based on the defined OFs and send them to the routers for advertisement. It uses different algorithms for rank calculation. We propose the Mesh TDMA Markov chain model [86] as an example tool to calculate the routers' rank. The scheme proposed in [86] is slightly modified in our work to adapt slot matrices and the results obtained from the Markov chain are used to calculate the routers rank. The model calculates the rank based on the routers uplink and downlink reliability and latency by considering the routing topology, link probabilities, and schedules in the network.

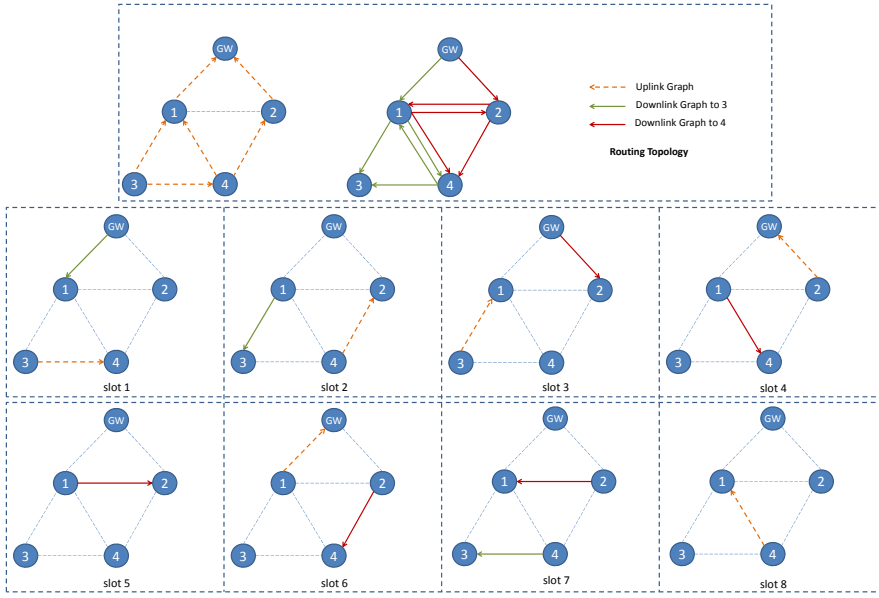


Figure 6.4: Multi-path routing example and their schedule

For example, to calculate the rank based on reliability, we build a probability matrix $p^{(t)}$ for each time slot t with p_{ij} being the probability of success of $link_{i \to j}$ and calculate the product matrix $P^{(t)} = p^{(1)} \dots p^{(t)}$. The cell c_{ij} in $P^{(t)}$ gives the probability of reaching node i from node j in t slots. In particular, c_{i0} the probability of reaching node i from Gateway (with id 0) can be considered as the rank of node i.

Figure 6.4 shows a sample network with four routers and a Gateway and shows how the communication is scheduled in time slots 1 to 8. For example, it

shows a link scheduled between the GW and router 1 in the downlink graph in timeslot 1 and in the uplink graph in timeslot 6.

For each slot t, we build a probability matrix in such a way that cell c_{ij} in matrix $p^{(t)}(i \neq j)$ is the probability assigned to (/success ratio of) link $i \rightarrow j$. A cell in the main diagonal (p_{ii}) gives the probability of staying at node i. This is 1 if the node is not scheduled for transmitting at slot t, otherwise it is the probability of failure $1 - p_{ij}$ of the scheduled link $i \rightarrow j$. We denote $q_{ij} = 1 - p_{ij}$. Gateway is node 0.

Probability matrices for the uplink graph, namely slots 1 – 8, are listed in the following matrices:

$$p^{(1)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & q_{34} & p_{34} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} p^{(2)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & p_{42} & 0 & q_{42} \end{pmatrix} \tag{6.1}$$

$$p^{(3)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & p_{31} & 0 & q_{31} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} p^{\{4)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ p_{20} & 0 & q_{20} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{6.2}$$

$$p^{(5)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} p^{(6)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ p_{10} & q_{10} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{6.3}$$

$$p^{(7)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} p^{(8)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & p_{41} & 0 & 0 & q_{41} \end{pmatrix} \tag{6.4}$$

Product matrices $p^{(1)}p^{(2)}$, $p^{(1)} \ldots p^{(3)}$, $p^{(1)} \ldots p^{(4)}$... and $P^{(8)} = p^{(1)}p^{(2)} \ldots p^{(8)}$ ($p^{(5)}$ and $p^{(7)}$ are identity matrix I; multiplication with identity matrix gives the same matrix: $I \times A = A \times I = A$, therefore multiplication is not needed)

$$p^{(1)}p^{(2)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & p_{34}p_{42} & q_{34} & p_{34}q_{42} \\ 0 & 0 & p_{42} & 0 & q_{42} \end{pmatrix} \tag{6.5}$$

$$p^{(1)}...p^{(3)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & q_{34}p_{31} & p_{34}p_{42} & q_{34}q_{31} & p_{34}q_{42} \\ 0 & 0 & p_{42} & 0 & q_{42} \end{pmatrix} \tag{6.6}$$

$$p^{(1)}...p^{(4)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ p_{20} & 0 & q_{20} & 0 & 0 \\ p_{34}p_{42}p_{20} & q_{34}p_{31} & p_{34}p_{42}q_{20} & q_{34}q_{31} & p_{34}q_{42} \\ p_{42}p_{20} & 0 & p_{42}q_{20} & 0 & q_{42} \end{pmatrix} \tag{6.7}$$

$$p^{(1)}...p^{(6)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ p_{10} & q_{10} & 0 & 0 & 0 \\ p_{20} & 0 & q_{20} & 0 & 0 \\ p_{34}p_{42}p_{20} + q_{34}p_{31}p_{10} & q_{34}p_{31}q_{10} & p_{34}p_{42}q_{20} & q_{34}q_{31} & p_{34}q_{42} \\ p_{42}p_{20} & 0 & p_{42}q_{20} & 0 & q_{42} \end{pmatrix} \tag{6.8}$$

The product matrix $P^{(8)} = p^{(1)}...p^{(8)}$ of the whole frame is

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ p_{10} & q_{10} & 0 & 0 & 0 \\ p_{20} & 0 & q_{20} & 0 & 0 \\ p_{34}p_{42}p_{20} + q_{34}p_{31}p_{10} & q_{34}p_{31}q_{10} + p_{34}q_{42}p_{41} & p_{34}p_{42}q_{20} & q_{34}q_{31} & p_{34}q_{42}q_{41} \\ p_{42}p_{20} & q_{42}p_{41} & p_{42}q_{20} & 0 & q_{42}q_{41} \end{pmatrix} \tag{6.9}$$

The cell c_{ij} of the product matrix $P^{(8)}$ gives the probability of reaching node j starting from node i in 8 slots of the frame. In the first column are the probabilities of reaching gateway (node number 0) from any other node 1, 2, 3 and 4, e.g. the probability of reaching Gateway (node 0) from node 3 in 8 time slots of the frame is $Prob(3 \Rightarrow 0) = p_{34}p_{42}p_{20} + q_{34}p_{31}p_{10}$ and from node 4 is

$Prob(4 \Rightarrow 0) = p_{42}p_{20}$. These values, $Prob(r - id \Rightarrow gateway - id)$, can be seen as ranks for routers in the uplink direction, where r-id is the router id and gw-id is the gateway id (set to 0). The same mechanism can be applied to calculate the downlink rank for each router.

6.5 Performance evaluation

In this section we evaluate the performance of ISA100.11a* when compared to the ISA100.11a standard in terms of metrics such as reliability, real time and power consumption that are critical for industrial applications. We also compare the communication schedules and the management efficiency of both approaches in different scenarios.

We simulated ISA100.11a and ISA100.11a* in the NS-2 network simulator. We assumed that each router has similar Sub-network Manager to manage its local star topology. The simulation model, parameters and other details are summarized in Table 6.3.

Table 6.3: NS-2 simulation parameters and values

Parameter	Value
Number of the nodes	1 Gateway, 2 access points 22 routers and 38 I/O
Simulation area	100×100 m^2
Routers placement	Regular distribution (in 4 rows & 4 hops)
I/O devices placement	Random distribution
Radio propagation model	Two-ray ground
Data rate	250 Kbps
Radio range	15 m
Frequency Band and channels	2.4 GHz, 11 - 26 channels
Sensor traffic rate	1 per 4 s
Application traffic model	Constant bitrate (CBR)
Management superframes	2 s

6.5.1 Reliability and Real Time Guarantee

To evaluate the reliability and real time guarantee of ISA100.11a* and ISA100.11a in the presence of external interferences, we dropped the link quality in the network and measure the packet delivery ratio. The packet delivery ratio is calculated based on the number of packets received at the Gateway/actuators for the CBR traffic (periodic sensor data) sent from sensors. In the first experiment,

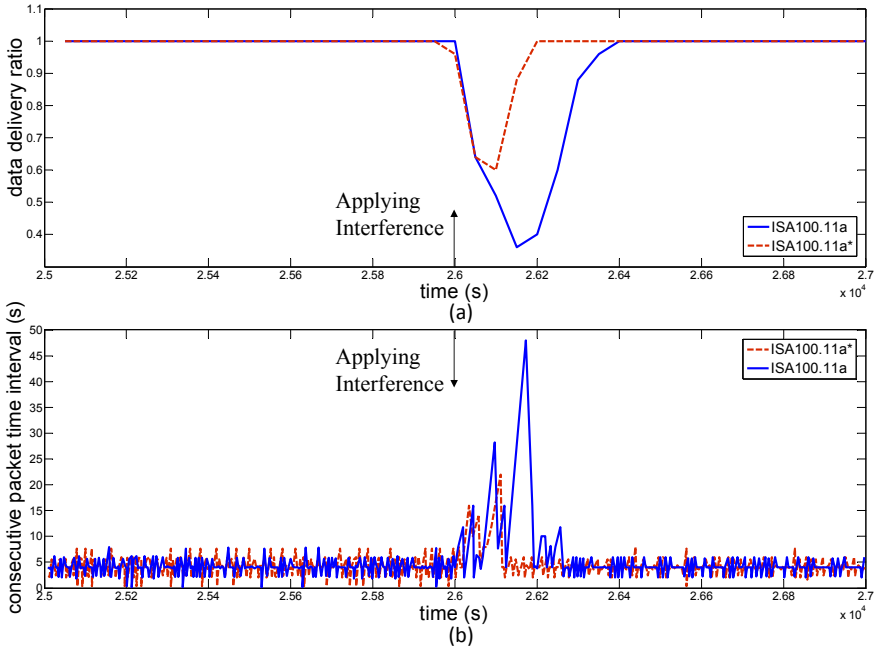


Figure 6.5: Reliability and real time evaluation

external interference is applied in the star sub-network between I/O devices and routers. Figure 6.5 (a) shows that the packet delivery ratio drops suddenly for both approaches, but it takes longer time for the standard approach to revert to the stable state. Figure 6.5 (b) shows that the jitter in the consecutive packet reception time-difference. It varies slightly from the expected value of 4 seconds (data traffic rate) in normal operations, but in the presence of interference the ISA100.11a requires longer duration than ISA100.11a* to reach back to the nor-

mal values. These two results show that the reliability and real time aspects can be improved with the proposed approach. The basic reason for the improvement is that in ISA100.11a it is the SM which performs repairs on receiving the periodic neighbor diagnostic reports causing more communications and delay, whereas in ISA100.11a*, the I/O devices can use their local statistics to fix the problem.

A second experiment has been done to measure the impact of hybrid management especially the rank advertisements on the performance. Here, the SM in both approaches deliberately attempts not to release the interfered communication links and not to use MAC re-transmissions. Now patterned link failures (interferences at small regions) are applied in the mesh network at different steps and the packet delivery ratio is measured.

Figure 6.6 (a), (b) and (c), show the variation in data delivery ratio of applying patterned failures in two small regions of the network in three different scenarios. They are applied in two steps (at an interval of 1,000 seconds) by changing the packet drop ratio from 50% to 70% and then to 80% in the three scenarios. ISA100.11a* outperforms ISA100.11a as it could improve the end-to-end reliability and reach a stable data-delivery-ratio much faster. This is because in ISA100.11a*, the I/O devices can re-select the best routers based on the new routers' ranks advertised, although the SM does not repair the interfered edges and routes in both approaches.

6.5.2 Communication Schedules

Figure 6.7 shows the global matrix of constructed schedules for 22 end-to-end connections with a publishing period of two seconds in ISA100.11a and ISA100.11a*. In ISA100.11a, the SM solely schedules interference-free cell and manages all allocations. There the distribution of allocated cells is more dense at the beginning of the superframe. In the extended ISA, a part of the superframe is managed by the SM but the rest is used by the routers to manage their local sub-network. Based on the I/O devices distribution, the number of I/O devices associated with each router and the traffic characteristics of I/O devices, the routers assign different amount of resources I/O devices.

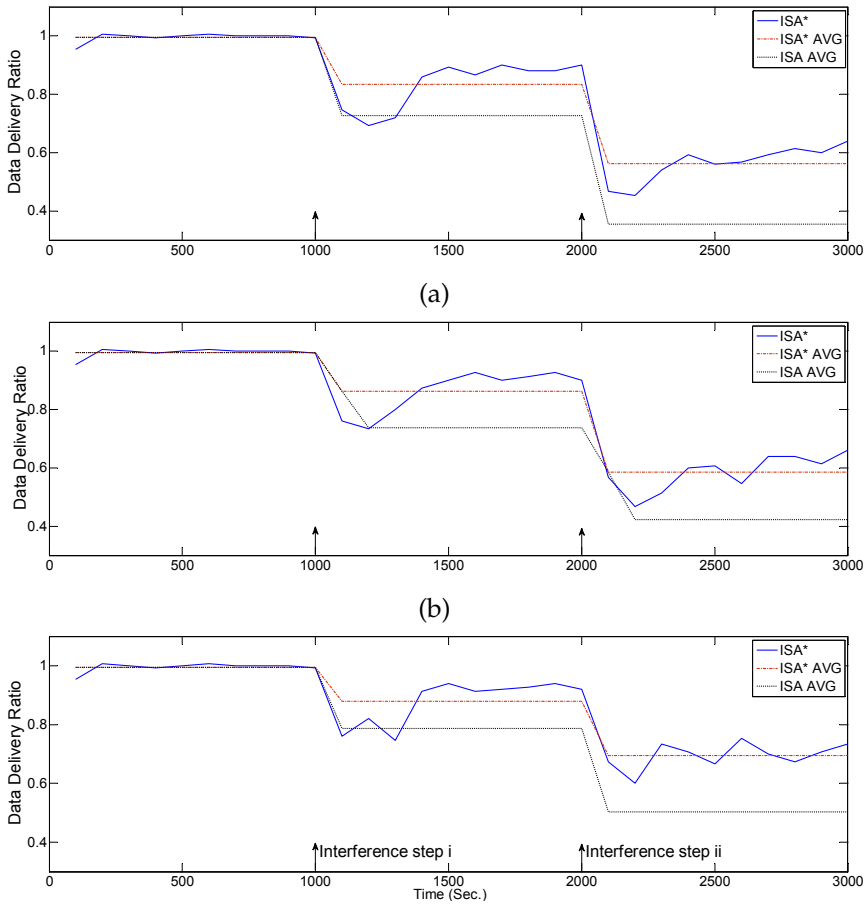


Figure 6.6: Data delivery ratio differences three scenarios; with (a) 80% packet drop ratio, (b) 70% packet drop ratio, and (c) 50% packet drop ratio

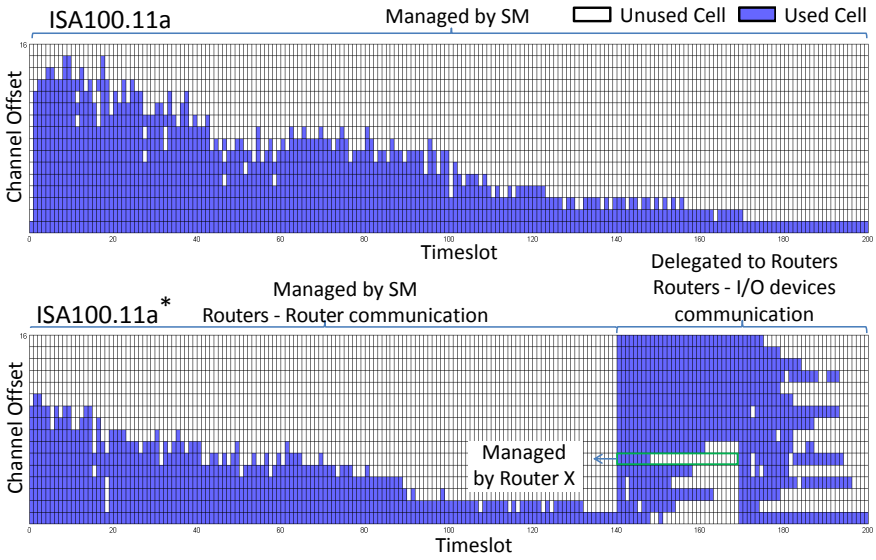


Figure 6.7: Slot-channel matrix for a sample network

6.5.3 Management Efficiency

6.5.3.1 Node joining process

To evaluate the I/O device joining process, we consider the overhead and delay of reserving management resources for both approaches. We do not consider the scanning delay before joining in this evaluation. The joining delay and communication overhead with hop distance are given in Figure 6.8 (a) and (b) respectively. As the hop distance increases, in traditional ISA the delay and communication overhead increases, whereas in ISA100.11a* they are more or less constant. Moreover, the delay and overhead of the proposed approach are much smaller than the tradition approach. This is because in traditional ISA, the routers forward the I/O device's join request to the SM to send the response and reserve communication resources, whereas in the proposed approach the routers themselves handle it locally. The results show that the proposed approach can performs far better than traditional approach in large-scale networks and in those scenarios where energy-harvested I/O devices joins and leaves the network frequently.

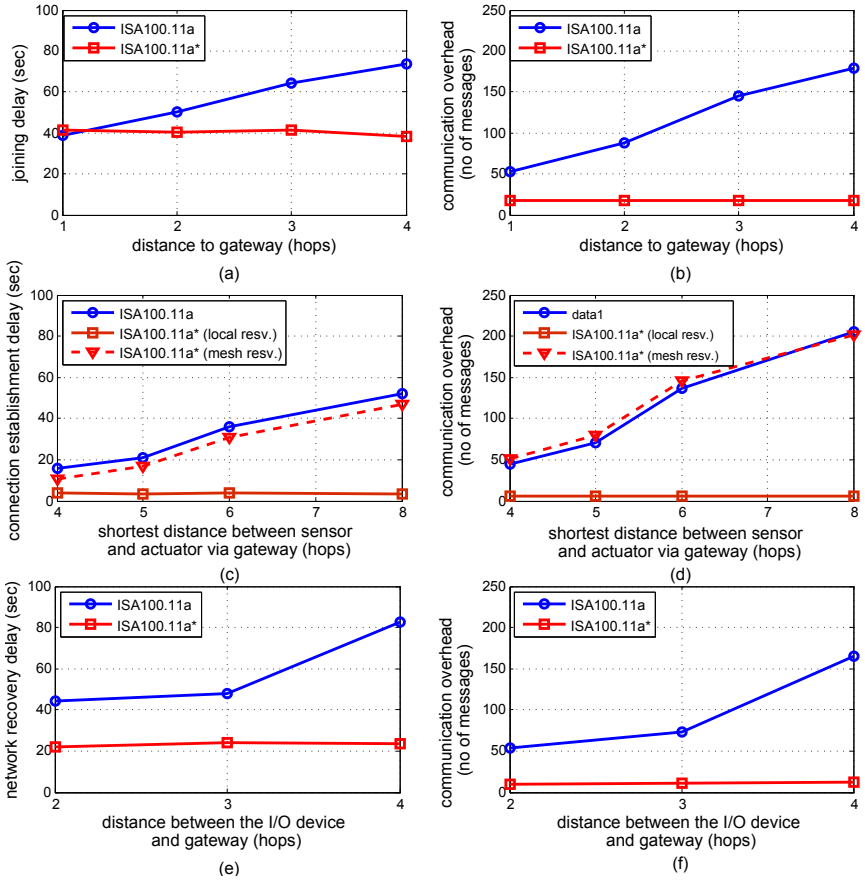


Figure 6.8: Evaluation of management efficiency

6.5.3.2 End-to-end connection establishment

To evaluate the management efficiency in end-to-end connection establishment, we measure the communication overhead and delay experienced for reserving the communication resources between the sensors and their final destinations (Gateway/actuators). In this experiment we disabled the overprovisioning policy so that no resources are readily available for the routers in the mesh

network to support I/O devices traffic requirements. We measure separately the communication overhead and delay for reserving the communication resources between the I/O devices and their selected routers (ISA* local reservation), and between the routers and Gateway in the extended approach (ISA* mesh reservation).

Figure 6.8 (c) and (d) displays, the results of the management efficiency of both approaches in end-to-end connection establishment. It is noticeable that the increase in the hop distance between sensor and their destination results in more delay and larger number of communications for establishing connection for both approaches, except for the local reservation of communication resources between I/O devices and their routers, where they remain almost constant. If we allow overprovisioning and resources are readily available in the mesh network, the overhead and delay of the extended approach come close these local reservation values.

6.5.3.3 Coping with changes and disturbances in the network

To evaluate the management efficiency in coping with changes and disturbances in the network, we introduce edge failures between the I/O devices and chosen routers and measure the number of required communications and delay for overcoming the failures. In the traditional ISA, such failures might results in sending connectivity alert to the SM which in turn configures new routers and resources to the I/O devices. In ISA100.11a*, the I/O device chooses a new router based on its OFs, sends joining request and use the allocated local resources of the router.

Figure 6.8 (e) and (f) shows the results of the experiments and it clearly shows that the localized management of the extended approach has much lower overhead (92% lesser at 4th hop) and delay (70% lesser at 4th hop) when compared to the centralized standard approach.

6.5.4 Power Consumption

To evaluate the energy-consumption of network nodes in ISA100.11a and ISA100.11a*, the simulation is run for 1,000 seconds. We followed the same equations and parameters given in [19] to calculate the energy consumption in terms of Tx/Rx turnaround (neglecting the processing energy). We consider two states of network operation, namely a static and a dynamic environment (e.g. link failures). In the static environment we measure the energy needed to

Table 6.4: Periodic messages in ISA100.11a and ISA100.11a*

Item	Parameter	Value	Transmission type
Periodic management data	Channel and neighbor diagnostics report	30 s	Acknowledged unicast
	Advertisement rate	4 s	Un-Acknowledged broadcast
Application Data	Sensor Data rate	4 s	Acknowledged unicast

Table 6.5: Energy-consumption in the network (in 1,000 s) during normal operation

Environment	Item		ISA100.11a	ISA100.11a*
Static	Network management energy		33.71 J	28.78 J
	Average router energy		4.32 J	3.82 J
	Average I/O device energy		0.36 J	0.34 J
	Total energy (without idle)		62.19 J	58.68 J
	Idle listening Energy		60.49 J	50.23 J
Dynamic (One edge failure)	Network maintenance energy	2 hop	0.033 J	0.006 J
		3 hop	0.044 J	0.006 J
		4 hop	0.105 J	0.008 J
	I/O device energy including idle listening	2 hop	0.0073 J	0.0064 J
		3 hop	0.0088 J	0.0068 J
		4 hop	0.0154 J	0.0086 J

exchange network management messages (periodic updates), as well as application data messages (from sensors to actuators). For the dynamic environment, we measure the energy consumed for the network maintenance.

The management and application data messages in ISA100.11a and ISA100.11a* are listed in Table 6.4. The total energy consumption of the network for management and application traffic is provided in Table 6.5 and we can see that it is almost equal in ISA100.11a and ISA100.11a*. The routers on average consume ten times more energy than the I/O devices in both approaches. Table 6.5 also lists the consumed energy by the I/O devices and for network maintenance messages, in case of edge failures at different hop distance from Gateway. ISA100.11a* has less overhead and less maintenance energy for coping with

disturbances (e.g., edge failures) in the network. For example, when the edge failures between an I/O device and the router happen at four hop distance from the Gateway, ISA100.11a* consumes 0.007 J to overcome the failure, whereas ISA100.11a requires 0.102 J. The I/O devices in ISA100.11a* consume less energy when compared to ISA100.11a, as they receive the join replies and communication resources from the new routers faster and hence spend less energy during idle listening.

6.6 Conclusion and future work

We have proposed ISA100.11a*, an extension to ISA100.11a standard, to better support the requirements of resource constrained I/O devices, to improve the scalability of the network (concerning the number of I/O devices supported) and to mitigate the problems of link changes in large-scale dynamic networks. We introduced a new hybrid network management scheme where part of the management responsibilities and the authority over communication resources are delegated to the routers. This improves management efficiency. The proposed enhancement also allows I/O devices to choose the best possible routers according to their desired metric, using local statistics as well as the advertised routers' ranks. This gives the I/O devices the flexibility to choose/change their routers, which improves efficiency and helps them cope better with link failures.

We compare the performance of ISA100.11a* with ISA100.11a in a typical industrial environment with high packet losses. We evaluate the reliability and real time aspects, power consumption, communication schedule and management efficiency of both approaches. We show that data delivery ratio and end-to-end delay can be improved in ISA100.11a* with lower power consumption. We also show that ISA100.11a* can achieve higher efficiency in network management in terms of latency and overhead during node joining, resource reservation, end-to-end connection establishment, and coping with dynamic situations.

We plan to showcase the working of ISA100.11a* in practice using the hardware platform developed in the EU FP7 project WiBRATE. We also aim at maintaining backward compatibility to the ISA100.11a standard so that it can operate in an already deployed ISA100.11a network. Although no security issues are foreseen, but for security key distribution from Security Manager to routers for device authentication during joining, further analysis is needed. The planned extension will be contributed back to the standardization body, so that it can be adopted by the industrial community.

CHAPTER 7

Conclusion

Monitoring and process control applications in industrial automation require real-time, reliable and low power wireless communication. This thesis has presented several distributed and hybrid management schemes that fulfill these requirements. In Chapter 2, we reviewed state-of-the-art solutions based on their strengths and drawbacks in addressing the main metrics in the industrial automation domain. In Chapter 3, we evaluated the WirelessHART standard. The outcomes of the WirelessHART evaluation are also applicable to ISA100.11a networks, due to their similarities in lower layers and network management. In Chapter 4, we introduced a distributed management scheme designed to address wireless industrial automation requirements, including for battery-powered I/O devices that can participate in the routing task and the communication scheduling task. In Chapter 5, the distributed management scheme was put forward as a means to fulfill the requirements of power constraint I/O devices (e.g., harvester-powered). In Chapter 6, we proposed an extension to the ISA100.11a standard to fulfill the requirements of power constraint I/O devices in harsh and dynamic industrial environments.

In Section 7.1, we first elaborate on the main contributions and results of this thesis to provide reliable and real-time wireless communication. Next, Section 7.2 revisits and answers our research questions. Finally, Section 7.3 provides directions for future work.

7.1 Contributions

The contributions of the thesis are revisited in the following:

(Contribution 1) Implementation and validation of the WirelessHART simulator in NS-2: WirelessHART, was introduced to address industrial process

automation and control requirements. The standard can be used as a reference point to evaluate other wireless protocols in the domain of industrial monitoring and control. This makes it worthwhile to set up a reliable WirelessHART simulator to achieve that reference point in a relatively easy manner. Chapter 3 explains our implementation of WirelessHART in the NS-2 simulator. According to our knowledge, this is the first implementation that supports the WirelessHART network manager as well as the whole stack of the WirelessHART standard. This implementation offers an alternative to expensive testbeds for evaluating WirelessHART. We evaluated the performance of our implementation in terms of delay and communication load in the network. We observed that WirelessHART cannot cope with dynamicity in the network in a real-time manner and that it incurs a high management overhead. As the network scales up, this problem is more exacerbated.

(Contribution 2) A Distributed Network Management Scheme for Real-Time Monitoring and Process Control Applications in Wireless Industrial Automation: In this contribution, we proposed a distributed network management scheme, namely D-MSR. It enables the network devices to join the network, to schedule their communications, to establish end-to-end connections by reserving communication resources to address real-time requirements, and to cope with network dynamicity (e.g., node/edge failures) in a distributed manner. According to our knowledge, this is the first distributed management scheme based on the IEEE 802.15.4e standard, which guides the nodes in different phases: from joining until publishing their sensor data in the network. We demonstrated via simulation that D-MSR can address real-time and reliable communication as well as the high throughput requirements of industrial automation wireless networks. It also achieved higher efficiency in network management than WirelessHART, both in terms of delay and overhead. In addition, we observed that the distributed management scheme in D-MSR can perform well as the network scales up. This scheme can cope locally with the dynamicity in a real-time manner.

(Contribution 3) A Distributed Management Scheme for Hybrid Networks to Provide Real-time Industrial Wireless Automation: We proposed a distributed management scheme named D-MHR, which can address the requirements of energy constrained I/O devices. Unlike in the D-MSR, in D-MHR, the routers can dynamically reserve communication resources and manage the I/O devices in the local star sub-networks. We demonstrated that D-MHR achieves higher network management efficiency compared to the ISA100.11a standard, without compromising the latency and reliability requirements of industrial wireless

networks. As in D-MSR, the distributed management capability of the routers helps D-MHR to perform well as the network scales up. This scheme can cope locally with the dynamicity in the network.

(Contribution 4) ISA100.11a*: The ISA100.11a extension for supporting energy-harvested I/O devices: We proposed an extension to ISA100.11a to better address the requirements of the energy constrained I/O devices. The proposed extension decentralizes the management by delegating a part of the management responsibility to the routers in the network. It also allows the I/O devices to choose their best possible routers based on various metrics, by considering the local statistics and advertised routers' ranks. We showed that the proposed extension solved the real-time and reliability requirements of industrial wireless networks more efficiently than the traditional ISA100.11a standard can do. Thanks to the capability of the routers to address the requirements of the I/O devices locally, ISA100.11a* can also achieve a higher network management efficiency than the ISA100.11a standard. Similar to D-MSR and D-MHR that use the distributed management schemes, the hybrid management scheme in ISA100.11a* performs well as the network scales up.

7.2 Conclusions

The main aim of this thesis was to address the (i) real-time and (ii) reliable communication requirements of periodic monitoring and process control applications in harsh and dynamic industrial environments. To that end, the main research question of this thesis was formulated as follows:

> *How to provide reliable and real-time communication to address the industrial wireless automation requirements of a harsh and dynamic industrial environment, while achieving higher efficiency in network management in terms of delay and overhead?*

To answer the research question, we evaluated the WirelessHART standard, as the first standard designed for the wireless sensor networks domain. WirelessHART was introduced to address industrial process automation and control requirements. This standard can be used as a reference point to evaluate other wireless protocols in the domain of industrial monitoring and control. The results of our WirelessHART evaluation also apply to ISA100.11a networks, due to similarities in lower layers and the network management. We found that the

network management algorithm greatly affects the performance of the WirelessHART network, namely during node joining, the connection establishment, data delivery latency, and when coping with node/link failure. Consequently, when applying other system management algorithms results may differ. We also observed that WirelessHART, by using the centralized management approach, incurs a high overhead to cope with dynamic situations in the network. The remainder of this thesis will therefore propose purely distributed or hybrid management schemes to mitigate those problems **(Contribution 1)**.

To answer the research question, we first assumed that I/O devices have sufficient power to participate in routing and communication scheduling tasks. Linked to this, we proposed a distributed management scheme, namely D-MSR, that supports the full mesh topology. D-MSR, which is a node-based scheme, tries to assign different communication resources to nodes in a two-hop neighborhood to avoid potential interferences and to increase network throughput. In order to provide real-time communication, we proposed a new distributed signaling protocol that reserves communication resources along the multi-path routes between the sensor and its final destinations. In order to provide reliable communication, we used multi-path routing and channel hopping schemes. The spatial reuse of communication resources in D-MSR improves the throughput in a large-scale network at the potential cost of reduced reliability due to internal interference. On the other hand, by avoiding the spatial reuse of communication resources in WirelessHART, the throughput is reduced. This makes WirelessHART less suitable for large-scale networks. The end-to-end delay in D-MSR is close to that of WirelessHART. This result shows that D-MSR can address real-time requirements, while also achieving a higher efficiency in network management than WirelessHART, in terms of delay and overhead **(Contribution 2)**.

In addition, we considered those applications in which the I/O devices are power-constraint nodes (e.g. Harvested-power devices). In these types of devices, energy availability varies in a non-deterministic manner. As a result, these types of devices cannot participate in routing and communication scheduling tasks. The node-based solution, such as employed by D-MSR, incurs scheduling overhead that may not be suitable for these types of devices in practice. As a result, we proposed a cluster-based solution that is more suitable for constrained power I/O devices. This scheme is a purely distributed management scheme that allocates the communication resources to the routers (cluster-heads) in a distributed manner to facilitate real-time communication. In D-MHR, we gave more capabilities to the routers and less to the I/O devices. Unlike the in D-MSR, in D-MHR, the routers are able to manage the I/O devices by forming local

sub-networks. The harvester powered I/O devices in this scheme can choose the best possible neighbor routers based on their requirements. As a result, D-MHR provided real-time and reliable communication in industrial wireless automation, while it also achieved higher efficiency in network management than ISA100.11a, in terms of delay and overhead.

Similar to D-MSR, D-MHR applies the spatial reuse of communication resources. Routers far away from each other (more than two-hops) can choose the same communication resources. This means that the same cell can be reused in several neighborhoods. On the other hand, in ISA100.11a, the central system manager schedules all the communication and there is no scope of re-using the dedicated cells in the network. The spatial reuse of communication resources (i.e. channel offsets) in D-MHR leaves parts of cells un-used. Therefore, the spatial reuse of communication resources in D-MHR helps to improve the network throughput in a large scale-network **(Contribution 3)**.

Finally, we proposed an extension to the ISA100.11a standard, which uses hybrid network management. Unlike D-MSR and D-MHR that use the purely distributed management approach, the hybrid management scheme in the proposed extension entails managing the mesh network between the routers (cluster-heads) in a centralized manner and managing the star sub-network in a distributed manner. This extension allocates communication resources to routers to address the requirements of I/O devices. Routers' ranks are calculated based on different OFs by the SM and are advertised by the routers to let I/O devices choose the best routers based on their requirements. As a result, the I/O devices join the network much faster, and re-select their routers using various metrics, by considering local statistics and routers' ranks **(Contribution 4)**.

In conclusion, this thesis has contributed to the existing literature on industrial wireless communication by providing new insights into monitoring and process control applications in wireless industrial automation. The solutions presented throughout the chapters have the potential to address the requirements of wireless industrial automation. While the applications, including battery-powered and harvested-power I/O devices, have distinct requirements, they both share crucial real-time and reliable requirements. As a result, by modifying the management scheme and using the resources reservation scheme, we can address the real-time requirements. In addition, by applying multi-path routing and channel hopping techniques reliable communication can be provided.

7.3 Future research directions

This thesis has addressed problems related to various distributed management schemes in wireless industrial automation. However, there are still issues to be addressed in future work.

- **Evaluating different network manager algorithms**: WirelessHART and ISA-100.11a use centralized network management techniques for communication scheduling and to construct routes. However, those standards do not specify the particular optimization algorithms that can be used by the network manager to allocate resources and to construct the routes. In Chapter 3, we found that the network management algorithm greatly affects the performance of the WirelessHART network, namely during node joining, the connection establishment, data delivery latency, and when coping with node/link failure. Consequently, when applying other network management algorithms results may differ. We intend to use different network manager algorithms in the simulator and evaluate the performance of those algorithms.

- **Supporting bursty traffic**: in this thesis, we used concepts from ATM networks to fulfill real-time requirements. While our present protocol solely focuses on constant bit-rate traffic, it would be interesting to extend it to support bursty traffic as well. Thus, the network can cope with the bursty nature of data traffic generated by the applications in the case of event occurrence when the large amount of traffic or reports needs to be forwarded to their destination. ATM already provides a solution for the delivery of bursty traffic over a shared network; it considers a virtual circuit with statistical multiplexing. A similar mechanism can be applied to D-SAR to support bursty traffic.

- **Distributed collaborative power control method**: in two sections of this thesis, we allow a receiving node to gather channel offset information about its two-hop neighborhood, and to choose a free channel offset based on this information. This scenario does not guarantee that the hidden terminal problem is solved, because even offset information from the two-hop neighborhood does not guarantee that two nodes that are in interference range do not transmit at the same time and hence cause collisions. In order to improve reliability, the nodes monitor the status of their communication on each cell and thanks to the scheduled communication concepts, this internal interference can be detected by observing the constant packet loss in those cells after reservation. Alternatively, it could be interesting to use distributed and collaborative

power control techniques to enable the node detecting interference to instruct the interfering node to re-adjust its transmission power to reduce interference.

- **Considering adaptive channel hopping (ACH) mechanism**: channel hopping is often used to mitigate external interference and multipath fading. In this thesis, we considered the blind channel hopping technique in the data link layer. The other solution is using the adaptive channel hopping (ACH) technique, in which the channel is changed on a link-by-link basis, but only when necessary. There is a tradeoff between using blind channel hopping and ACH. In the former, if the node switches to another congested channel or switches from a good channel to a congested one, this hopping does not help to mitigate the interference and just wastes energy [33]. However in ACH, nodes only change their frequencies when interference is detected on the current operating channel. Using ACH instead of considering blind channel hopping can be helpful. However, nodes need to collaborate to be able to decide which channel to switch to. This can introduce a significant overhead, since nodes need to continuously scan all channels for interference levels. Furthermore, nodes need to ensure that while a communicating pair chooses the same frequency, neighboring pairs use different frequencies. It would therefore be interesting to add ACH to the data link layer in the future.

- **In-Network Data Aggregation for Control Operations**: two types of aggregation, data aggregation and packet aggregation, are supported by WIA-PA in order to reduce the number of packet transmissions. WirelessHART, WISA, and ZigBee Pro do not support this function. In certain industrial closed-loop control applications involving multiple sensors and an actuator, raw sensor readings are streamed from the sensors to the actuator. The actuator subsequently performs computations using the readings to carry out the relevant control operations. This traditional approach, however, is not suitable for multihop wireless sensor networks, since they have highly limited bandwidths. The idea would then be to allow an intermediate node to carry out the computations and only send the final control output to the actuator, thus saving network bandwidth. However, as every actuation operation may be dependent on a different set of sensors, the nodes need to autonomously decide which node should act as the intermediate aggregation node that will be responsible for computing the control output. This technique will also contribute towards improving real-time operation.

- **Applying MIMO and OFDM in Physical Layer**: over the last decade, multiple antenna and Multiple Input Multiple Output (MIMO) techniques have

been widely discussed to increase the reliability and throughput of various wireless systems. The antenna diversity can also improve reliability by achieving multiple different realizations of channel. So, if one antenna gets an interfered/distorted signal, which is non-recoverable due to signal propagation through a deep faded channel, another antenna may receive a copy of the signal which is suitable for decoding. This can significantly improve link reliability. MIMO can increase the system throughput without increasing the bandwidth. Several antennas to transmit/receive a portion of a signal with spatial diversity make this possible. Orthogonal Frequency Division Multiplexing (OFDM) takes this to the next step by using orthogonal sub-carriers in a frequency band. OFDM not only increases the throughput of a system, but also enables the facility to use different levels of modulation for different sub-carriers, based on the channel state information (CSI). Such techniques have been applied in the recently developed IEEE 802.11n standard to increase WiFi throughput to a next level [87]. However, the complexity of the receivers also increases to facilitate this technology, which makes it challenging to implement in WSN applications. Building simple MIMO transceivers with a low power consumption is still in a research phase [88].

- **Neighbor discovery in a multi-channel network**: due to channel hopping and multichannel communication, the process of node joining and neighbor discovery are challenging issues [89]. Another issue is the scheduling of broadcasting links in a distributed manner. It might be worthwhile to propose a scheme that can either help the node (i) during joining or (ii) during neighbor discovery to discover it potential neighboring nodes. As a result, nodes can (re)join the network faster and they can cope with node and link failure more efficient, even in case of a dynamic and harsh industrial environment.

- **Implementation of proposed schemes in real-hardware**: we plan to showcase the working of ISA100.11a* in practice, by using the hard-ware platform developed in the EU FP7 project WiBRATE.

Bibliography

[1] *Fieldbus Foundation, "Technical overview," [Online]. Available: http://www.fieldbus.org*.

[2] C. Internationa, "Controlnet specification," 1999.

[3] I. E. Commission *et al.*, "Iec 61158: Digital data communications for measurement and control-fieldbus for use in industrial control systems," 2003.

[4] I. Standard, "11898: Road vehicles—interchange of digital information—controller area network (can) for high-speed communication," *International Standards Organization, Switzerland*, 1993.

[5] J. Berge, *Fieldbuses for Process Control: Engineering, Operation, and Maintenance*. ISA, 2001.

[6] A. Willig, K. Matheus, and A. Wolisz, "Wireless technology in industrial networks," *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1130–1151, 2005.

[7] K. Pister, P. Thubert *et al.*, "Industrial routing requirements in low-power and lossy networks," 2009.

[8] J. Frey and T. Lennvall, *Embedded Systems Handbook: Networked Embedded Systems*. CRC PressI Llc, 2009, vol. 6, ch. Wireless Sensor Networks for Automation, pp. 21–43.

[9] *ZigBee PRO specification*, ZigBee Alliance Std., Oct. 2007.

[10] D. Christin, P. S. Mogre, and M. Hollick, "Survey on wireless sensor network technologies for industrial automation: The security and quality of service perspectives," *Future Internet*, vol. 2, no. 2, pp. 96–125, 2010. [Online]. Available: http://www.mdpi.com/1999-5903/2/2/96

[11] R. Wagner and R. Barton, "Performance comparison of wireless sensor network standard protocols in an aerospace environment: Isa100.11a and zigbee pro," in *Aerospace Conference, 2012 IEEE*, March 2012, pp. 1–14.

[12] *Industrial communication networks – Fieldbus specifications – Wireless systems for industrial automation: process control and related applications*, IEC/PAS Std. 62 734, Rev. Edition 1.0, Mar. 2012.

[13] *Industrial communication networks - Fieldbus specifications, WirelessHART communication network and communication profile*, IEC/PAS Std. 62 591, Rev. Ed. 1.0., 2009.

[14] E. Lattanzi, E. Regini, A. Acquaviva, and A. Bogliolo, "Energetic sustainability of routing algorithms for energy-harvesting wireless sensor networks," *Comput. Commun.*, vol. 30, pp. 2976–2986, 2007.

[15] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*, Internet Engineering Task Force (IETF) Std. 2070-1721, Mar. 2012.

[16] P. Zand, S. Chatterjea, K. Das, and P. Havinga, "Wireless industrial monitoring and control networks: The journey so far and the road ahead," *Journal of Sensor and Actuator Networks*, vol. 1, no. 2, pp. 123–152, 2012. [Online]. Available: http://www.mdpi.com/2224-2708/1/2/123

[17] T. Watteyne, S. Lanzisera, A. Mehta, and K. Pister, "Mitigating multipath fading through channel hopping in wireless sensor networks," in *Communications (ICC), 2010 IEEE International Conference on*, 2010, pp. 1–5.

[18] *IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY)*, IEEE Computer Society Std., Rev. Revision of IEEE Std 802.15.4-2003, Sep. 2006.

[19] P. Zand, A. Dilo, and P. Havinga, "D-msr: A distributed network management scheme for real-time monitoring and process control applications in wireless industrial automation," *Sensors*, vol. 13, no. 7, pp. 8239–8284, 2013. [Online]. Available: http://www.mdpi.com/1424-8220/13/7/8239

[20] ——, "Implementation of wirelesshart in ns-2 simulator," in *Emerging Technologies Factory Automation (ETFA), 2012 IEEE 17th Conference on*, Sept 2012, pp. 1–8.

[21] P. Zand, E. Mathews, P. Havinga, S. Stojanovski, E. Sisinni, and P. Ferrari, "Implementation of wirelesshart in the ns-2 simulator and validation of its correctness," *Sensors*, vol. 14, no. 5, pp. 8633–8668, 2014. [Online]. Available: http://www.mdpi.com/1424-8220/14/5/8633

[22] P. Zand, S. Chatterjea, J. Ketema, and P. Havinga, "A distributed scheduling algorithm for real-time (D-SAR) industrial wireless sensor and actuator networks," in *IEEE 17th Conference on Emerging Technologies Factory Automation (ETFA)*, 2012, pp. 1–4.

[23] P. Zand, K. Das, E. Mathews, and P. Havinga, "D-mhr: A distributed management scheme for hybrid networks to provide real-time industrial wireless automation," in *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Accepted for Publication, 2014.

[24] ——, "A distributed management scheme for supporting energy-harvested i/o devices," in *IEEE 19th Conference on Emerging Technologies Factory Automation (ETFA)*, Accepted for Publication, 2014.

[25] P. Zand, E. Mathews, K. Das, A. Dilo, and P. Havinga, "Isa100.11a*: The isa100.11a extension for supporting energy-harvested i/o devices," in *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Accepted for Publication, 2014.

[26] V. Gungor and G. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *Industrial Electronics, IEEE Transactions on*, vol. 56, no. 10, pp. 4258–4265, 2009.

[27] G. Scheible, D. Dzung, J. Endresen, and J.-E. Frey, "Unplugged but connected [design and implementation of a truly wireless real-time sensor/actuator interface]," *Industrial Electronics Magazine, IEEE*, vol. 1, no. 2, pp. 25–34, 2007.

[28] *Industrial communication networks - Fieldbus specifications - WIA-PA communication network and communication profile*, IEC/PAS Std. 62 601, Rev. Edition 1.0, Nov. 2011.

[29] *IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer*, Std., 2012.

[30] K. Pister and L. Doherty, "Tsmp: Time synchronized mesh protocol," *IASTED Distributed Sensor Networks*, pp. 391–398, 2008.

[31] I. ISO, "11898-1: 2003-road vehicles–controller area network," *International Organization for Standardization, Geneva, Switzerland*, 2003.

[32] A. Willig, "Recent and emerging topics in wireless industrial communications: A selection," *Industrial Informatics, IEEE Transactions on*, vol. 4, no. 2, pp. 102–124, 2008.

[33] J. Ortiz and D. Culler, "Multichannel reliability assessment in real world wsns," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, ser. IPSN '10. New York, NY, USA: ACM, 2010, pp. 162–173. [Online]. Available: http://doi.acm.org/10.1145/1791212.1791233

[34] B. Kerkez, T. Watteyne, M. Magliocco, S. Glaser, and K. Pister, "Feasibility analysis of controller design for adaptive channel hopping," in *Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools*, ser. VALUETOOLS '09. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009, pp. 76:1–76:6. [Online]. Available: http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7934

[35] "Ieee standard for information technology–local and metropolitan area networks–specific requirements–part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications - amendment 8: Medium access control (mac) quality of service enhancements," *IEEE Std 802.11e-2005 (Amendment to IEEE Std 802.11, 1999 Edition (Reaff 2003)*, pp. 1–212, 2005.

[36] A. Forum, *ATM User Network Interface (UNI) Specification Version 3.1*, 3rd ed. Prentice Hall, 1995.

[37] S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He, "Atpc: Adaptive transmission power control for wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, ser. SenSys '06. New York, NY, USA: ACM, 2006, pp. 223–236. [Online]. Available: http://doi.acm.org/10.1145/1182807.1182830

[38] G. Hackmann, O. Chipara, and C. Lu, "Robust topology control for indoor wireless sensor networks," in *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, ser. SenSys '08. New York, NY, USA: ACM, 2008, pp. 57–70. [Online]. Available: http://doi.acm.org/10.1145/1460412.1460419

[39] S. Han, X. Zhu, A. Mok, D. Chen, and M. Nixon, "Reliable and real-time communication in industrial wireless mesh networks," in *17th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, 2011, pp. 3–12.

[40] H. Zhang, P. Soldati, and M. Johansson, "Optimal link scheduling and channel assignment for convergecast in linear wirelesshart networks," in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, 2009. WiOPT 2009. 7th International Symposium on*, 2009, pp. 1–8.

[41] P. Soldati, H. Zhang, and M. Johansson, "Deadline-constrained transmission scheduling and data evacuation in wirelesshart networks," in *The European Control Conference 2009 (ECC'09)*, 2009.

[42] H. Zhang, P. Soldati, and M. Johansson, "Efficient link scheduling and channel hopping for convergecast in wirelesshart networks," *School of Electrical Engineering, Royal Institute of Technology (KTH), Tech. Rep*, 2009.

[43] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "Real-time scheduling for wirelesshart networks," in *Real-Time Systems Symposium (RTSS), 2010 IEEE 31st*, Nov 2010, pp. 150–159.

[44] S. Zhang, G. Zhang, A. Yan, Z. Xiang, and T. Ma, "A highly reliable link scheduling strategy for wirelesshart networks," in *Advanced Technologies for Communications (ATC), 2013 International Conference on*, Oct 2013, pp. 39–43.

[45] G. Fiore, V. Ercoli, A. Isaksson, K. Landernas, and M. Di Benedetto, "Multihop multichannel scheduling for wireless control in wirelesshart networks," in *Emerging Technologies Factory Automation, 2009. ETFA 2009. IEEE Conference on*, Sept 2009, pp. 1–8.

[46] D. Chen, M. Nixon, and A. Mok, *WirelessHART: Real-Time Mesh Network for Industrial Automation*. Springer, 2010.

[47] *URL: http://www.isi.edu/nsnam/ns*.

[48] M. Nobre, I. Silva, L. Guedes, and P. Portugal, "Towards a wirelesshart module for the ns-3 simulator," in *Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on*, 2010, pp. 1–4.

[49] K. Pister and L. Doherty, "TSMP: Time synchronized mesh protocol," *IASTED Distributed Sensor Networks*, pp. 391–398, 2008.

[50] C. De Dominicis, P. Ferrari, A. Flammini, E. Sisinni, M. Bertocco, G. Giorgi, C. Narduzzi, and F. Tramarin, "Investigating wirelesshart coexistence issues through a specifically designed simulator," in *Instrumentation and Measurement Technology Conference, 2009. I2MTC '09. IEEE*, 2009, pp. 1085–1090.

[51] OMNeT++, "http://www.omnetoo.org."

[52] M. De Biasi, C. Snickars, K. Landernas, and A. Isaksson, "Simulation of process control with wirelesshart networks subject to clock drift," in *Computer Software and Applications, 2008. COMPSAC '08. 32nd Annual IEEE International*, 2008, pp. 1355–1360.

[53] K. Shah, T. Seceleanu, and M. Gidlund, "Design and implementation of a wirelesshart simulator for process control," in *Industrial Embedded Systems (SIES), 2010 International Symposium on*, 2010, pp. 221–224.

[54] P. Ferrari, A. Flammini, M. Rizzi, and E. Sisinni, "Improving simulation of wireless networked control systems based on wirelesshart," *Computer Standards & Interfaces*, vol. 35, no. 6, pp. 605 – 615, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0920548913000251

[55] S. Han, X. Zhu, A. Mok, D. Chen, and M. Nixon, "Reliable and real-time communication in industrial wireless mesh networks," in *Real-Time and Embedded Technology and Applications Symposium (RTAS), 2011 17th IEEE*, 2011, pp. 3–12.

[56] *URL: www.linear.com/docs/4188*.

[57] W. Dai, "Crypto++ library," 2007.

[58] S. Raza, A. Slabbert, T. Voigt, and K. Landernas, "Security considerations for the wirelesshart protocol," in *Emerging Technologies Factory Automation, 2009. ETFA 2009. IEEE Conference on*, Sept 2009, pp. 1–8.

[59] S. Han, X. Zhu, K. Aloysius, M. Nixon, T. Blevins, and D. Chen, "Control over wirelesshart network," in *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*, Nov 2010, pp. 2114–2119.

[60] D. Conzonato, C. D. Odoardo, E. Bartaloni, L. Guidi, and D. Pestonesi, "The fieldbus technology in the new power plants of enel produzione," in *Real-Time and Embedded Technology and Applications Symposium (RTAS), 2011 17th IEEE*, 2011, pp. 3–12.

[61] *URL: http://www.flexipanel.com*.

[62] S. Han, J. Song, X. Zhu, A. Mok, D. Chen, M. Nixon, W. Pratt, and V. Gondhalekar, "Wi-htest: Compliance test suite for diagnosing devices in real-time wirelesshart network," in *Real-Time and Embedded Technology and Applications Symposium, 2009. RTAS 2009. 15th IEEE*, 2009, pp. 327–336.

[63] T. Rappaport and S. Sandhu, "Radio-wave propagation for emerging wireless personal-communication systems," *Antennas and Propagation Magazine, IEEE*, vol. 36, no. 5, pp. 14–24, Oct 1994.

[64] K. Sohrabi, B. Manriquez, and G. Pottie, "Near ground wideband channel measurement in 800-1000 mhz," in *Vehicular Technology Conference, 1999 IEEE 49th*, vol. 1, Jul 1999, pp. 571–574 vol.1.

[65] *URL: http://cds.linear.com/docs/en/datasheet/5900whmf.pdf*.

[66] L. Badia, A. Erta, L. Lenzini, and M. Zorzi, "A general interference-aware framework for joint routing and link scheduling in wireless mesh networks," *Network, IEEE*, vol. 22, no. 1, pp. 32–38, 2008.

[67] P. Gupta and P. Kumar, "The capacity of wireless networks," *Information Theory, IEEE Transactions on*, vol. 46, no. 2, pp. 388–404, 2000.

[68] S. Munir, S. Lin, E. Hoque, S. Nirjon, J. A. Stankovic, and K. Whitehouse, "Addressing burstiness for reliable communication and latency bound generation in wireless sensor networks," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*. ACM, 2010, pp. 303–314.

[69] P. Suriyachai, J. Brown, and U. Roedig, "Time-critical data delivery in wireless sensor networks," in *Distributed Computing in Sensor Systems*. Springer, 2010, pp. 216–229.

[70] M. Salajegheh, H. Soroush, and A. Kalis, "Hymac: Hybrid tdma/fdma medium access control protocol for wireless sensor networks," in *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, 2007, pp. 1–5.

[71] S. Ergen and P. Varaiya, "Pedamacs: power efficient and delay aware medium access protocol for sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 5, no. 7, pp. 920–930, 2006.

[72] L. F. van Hoesel and P. Havinga, "A lightweight medium access protocol (lmac) for wireless sensor networks: Reducing preamble transmissions and transceiver state switches," 2004.

[73] A. Tinka, T. Watteyne, and K. Pister, "A decentralized scheduling algorithm for time synchronized channel hopping," in *Ad Hoc Networks*. Springer, 2010, pp. 201–216.

[74] P. Zand, S. Chatterjea, J. Ketema, and P. Havinga, "D-sar: A distributed scheduling algorithm for real-time, closed-loop control in industrial wireless sensor and actuator networks," 2011.

[75] J. Martocci, M. Goyal, M. Philipp, A. Brandt, and E. Baccelli, "Reactive discovery of point-to-point routes in low power and lossy networks," 2013.

[76] Z. A. Eu, H.-P. Tan, and W. K. G. Seah, "Opportunistic routing in wireless sensor networks powered by ambient energy harvesting," *Comput. Netw.*, vol. 54, no. 17, pp. 2943–2966, Dec. 2010.

[77] S. Petersen and S. Carlsen, "WirelessHART versus isa100.11a: The format war hits the factory floor," *IEEE Industrial Electronics Magazine*, vol. 5, no. 4, pp. 23–34, 2011.

[78] Y. Wu, J. A. Stankovic, T. He, and S. Lin, "Realistic and efficient multi-channel communications in wireless sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*. IEEE, 2008.

[79] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic, and T. F. Abdelzaher, "Mmsn: Multi-frequency media access control for wireless sensor networks." in *Infocom*, vol. 6, 2006, pp. 1–13.

[80] O. D. Incel, L. van Hoesel, P. Jansen, and P. Havinga, "MC-LMAC: A multi-channel MAC protocol for wireless sensor networks," *Ad Hoc Netw.*, vol. 9, no. 1, pp. 73–94, Jan. 2011.

[81] Y. Kim, H. Shin, and H. Cha, "Y-mac: An energy-efficient multi-channel mac protocol for dense wireless sensor networks," in *Proceedings of the 7th international conference on Information processing in sensor networks*. IEEE Computer Society, 2008, pp. 53–63.

[82] X. Chen, P. Han, Q.-S. He, S.-l. Tu, and Z.-L. Chen, "A multi-channel mac protocol for wireless sensor networks," in *Computer and Information Technology, 2006. CIT'06. The Sixth IEEE International Conference on*. IEEE, 2006, pp. 224–224.

[83] A. Gupta, C. Gui, and P. Mohapatra, "Exploiting multi-channel clustering for power efficiency in sensor networks," in *Communication System Software and Middleware, 2006. Comsware 2006. First International Conference on*. IEEE, 2006, pp. 1–10.

[84] R. Diestel, *Graph Theory*, 3rd ed. Springer, 2005.

[85] "IEEE standard for local and metropolitan area networks–part 15.4: Low-rate wireless personal area networks (lr-wpans) amendment 1: Mac sublayer," *IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1 –225, 16 2012.

[86] P. Chen and S. Sastry, "Latency and connectivity analysis tools for wireless mesh networks," in *Proceedings of the 1st International Conference on Robot Communication and Coordination*, ser. RoboComm '07, 2007, pp. 33:1–33:8.

[87] J.-H. Lee, M.-S. Baek, and H.-K. Song, "Efficient mimo receiving technique in ieee 802.11n system for enhanced services," *Consumer Electronics, IEEE Transactions on*, vol. 53, no. 2, pp. 344–349, 2007.

[88] M. R. Ahmad, E. Dutkiewicz, and X. Huang, "Ber-delay characteristics analysis of ieee 802.15. 4 wireless sensor networks with cooperative mimo," in *Applied Electromagnetics, 2007. APACE 2007. Asia-Pacific Conference on*. IEEE, 2007, pp. 1–5.

[89] A. Gonga, T. Charalambous, and M. Johansson, "Neighbor discovery in multichannel wireless clique networks: An epidemic approach," in *Mobile Ad-Hoc and Sensor Systems (MASS), 2013 IEEE 10th International Conference on*, Oct 2013, pp. 131–135.

About the author

Pouria Zand obtained his B.Sc. degree in Electrical Engineering from the K.N.Toosi University of Technology (K.N.Toosi), Iran, in 2005. Later, he obtained his M.Sc. degree in Telecommunication Engineering from the University of Tehran, Iran, in 2008. In April of 2010, he joined the Pervasive Systems group at the University of Twente to pursue his Ph.D. degree in the field of wireless dissemination in industrial environments. During this time, he worked on the WiBRATE research project. In April of 2014, he becomes a postdoc researcher in the Pervasive Systems group at the University of Twente.

List of publications in which he participated in reverse chronological order:

1) P. Zand, E. Mathews, K. Das, A. Dilo, and P. Havinga, *ISA100.11a*: The ISA100.11a extension for supporting energy-harvested I/O devices*. In: WoWMoM 2014, Accepted for publication.

2) P. Zand, K. Das, E. Mathews, and P. Havinga. *D-MHR: A Distributed Management Scheme for Hybrid Networks to Provide Real-time Industrial Wireless Automation*. In: WoWMoM 2014, Accepted for publication.

3) P. Zand, A. Dilo, and P. Havinga. *D-MSR: A distributed network management scheme for real-time monitoring and process control applications in wireless industrial automation*. In: Sensors, vol. 13, no. 7,pp. 8239–8284, 2013.

4) P. Zand, A. Dilo, and P. Havinga. *Implementation of WirelessHART in NS-2 simulator*. In: IEEE 17th Conference on Emerging Technologies Factory Automation (ETFA), 2012, pp. 1–8.

5) P. Zand, S. Chatterjea, J. Ketema, P. Havinga. *A Distributed Scheduling Algorithm for Real-Time (D-SAR) Industrial Wireless Sensor and Actuator Networks*. In Proceedings of the 2012 IEEE 17th Conference on Emerging Technologies & Factory Automation (ETFA), Krakow, Poland, 17–21 September 2012; pp. 1–4.

6) P. Zand, S. Chatterjea, K. Das, P. Havinga. *Wireless industrial monitoring and control networks: The journey so far and the road ahead*. J. Sens. Actuat. Netw. 2012, 1, 123–152.

7) P. Zand, E. Mathews, P. Havinga, S. Stojanovski, E. Sisinni, and P. Ferrari, "Implementation of wirelesshart in the ns-2 simulator and validation of its correctness," Sensors, vol. 14, no. 5, pp. 8633–8668, 2014.

8) P. Zand, K. Das, E. Mathews, and P. Havinga, "A Distributed Management Scheme for supporting energy-harvested I/O devices," ETFA 2014, Accepted for publication.