DOI: 10.48011/asba.v2i1.1284

Development of a Network Manager Compatible with *Wireless*HART Standard ⋆

Gustavo Cainelli * **Max Feldman** * **Tiago Rodrigo Cruz,** *
Ivan Muller * **Carlos Eduardo Pereira** *

* *Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre, Rio Grande do Sul, Brasil (e-mail: gustavo.cainelli@gmail.com, max.feldman@ufrgs.br, tiagorcruz1@gmail.com, cpereira@ece.ufrgs.br, ivan.muller@ufrgs.br).*

Abstract: The use of industrial wireless networks has been growing continuously and it has become an alternative to wired networks. One of the main elements of an industrial wireless network is the network manager, this component is responsible for tasks related to the network construction and maintenance. This work presents the development of a network manager compatible with the *Wireless*HART protocol, but also customizable, where it is possible to make modifications in order to carry out studies with this protocol. Case studies are presented where the developed tool was used for studies related to communications scheduling, adaptive channel mapping and fast data collection, thus proving the efficiency of the proposed manager.

Keywords: Industrial Wireless Network, *Wireless*HART, Network Manager.

1. INTRODUCTION

The *Wireless*HART (WH) protocol provides secure and reliable communication complying with the industrial applications requirements, being one of the main candidates to provide wireless communication in process automation applications.

Figure 1 shows a typical WH network composed of a set of wireless field devices (sensors or actuators) connected to a gateway through access points (AP). All field devices have message routing capability, thus forming a mesh network. The gateway is responsible for connecting the wireless network to the automation plant. Along with the gateway is the Network Manager (NM) which is responsible for managing the network, provisioning new devices, configuring network parameters, routing and scheduling transmissions. The management in a WH network is centralized and thus allows the field devices hardware and software simplification, since the central point has knowledge of all the parameters of the network.

It is possible to find some commercial network managers developed by manufacturers such as Emerson, Endress Hauser and Phoenix Contact. The WH standard allows the manufacturer to define some different management techniques, such as routing and scheduling algorithms. Once a commercial solution is purchased, only small changes can be made in the network parameters, and management algorithms. Configurations related to routing and scheduling techniques usually cannot be changed on commercial equipment.

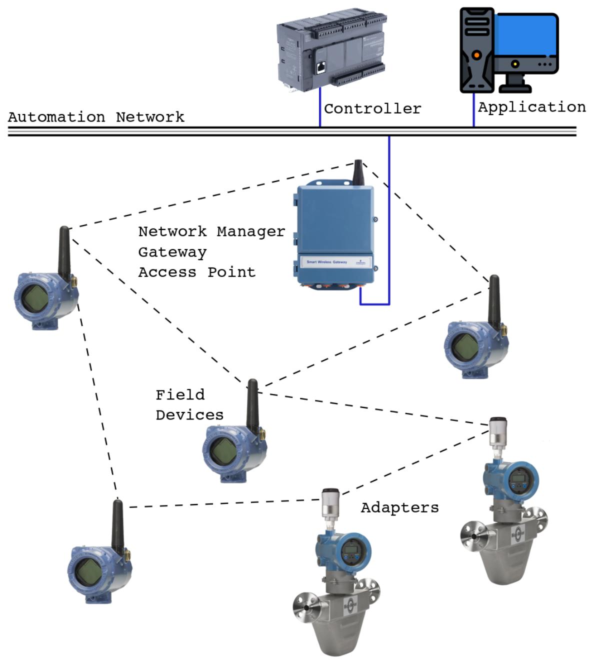

Fig. 1. *Wireless*HART mesh network topology.

Several works study algorithms related to management, or even propose protocol improvements, however, since it is not possible to change these algorithms in a commercial equipment, most studies perform simulated tests. Thus, having access to the source code of a network manager

⋆ This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 and by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

can be very useful for the study of industrial wireless communication protocols.

To implement new management algorithms and also for possible improvements in the protocol to be tested in a real network, it is necessary to have access to the equipment source code. Thus, the development of a network manager is necessary for the advancement of studies in industrial wireless networks (IWN).

In order to attend the basic demands of a WH network, the NM needs to have some minimum functions. The manager must configure the access point to propagate advertisement packets, provision new devices that listen to these packets and respond devices service requests.

The developed NM brings several possibilities of studies that explore new management algorithms and propose new techniques to improve the performance of industrial wireless networks. In this paper, three works already develop with this manager will be presented. One of them implemented a scheduling algorithms in the NM, another one created an adaptive channel mapping technique and the third one implemented a fast data collection technique. The rest of this paper is organized as follows. The Section 2 presents related works. The Section 3 describes the network manager. The Section 4 presents the hardware used. The Section 5 describes the results obtained in the works where the manager was used. Finally, the Section 6 presents the conclusions and future work.

2. RELATED WORKS

The WH communication protocol emerged as the first standard for wireless sensor network communications specifically for applications in process automation in the industrial environment Chen et al. (2010). The standard defines a series of specifications that must be implemented to meet requirements of industrial communications such as robustness and security. Since its first version, the standard has gone through several revisions that seek to clarify definitions and add or change specifications in the protocol. There are concepts in the standard that are not presented with many details of implementation, leaving the developer to choose the most appropriate way to do it. Some examples are: routing process, scheduling process and choice of security parameters for checking the reliability of new devices. These gaps become objects of study for researchers and concentrate most of the works related to this subject.

In Sanchez (2011) a study on the design and architecture of a manager for WH networks is presented. The author presents a review of the specifications of the WH standard focusing on the operations and features of the NM. Solutions are proposed for issues that are left open or that do not have specific implementation details regarding the NM. Case studies are presented, with details of essential process necessary for the network operation, such as network initialization, device aggregation, device service request, health report, alarm report and network update.

In Hahn (2011), an access point for WH networks was developed. The Network Access Point (NAP) was divided into two parts. The first, called NAP-Host, is an entity that runs on Linux and has the general objective of making the connection between the physical access point and the gateway. Upon initialization, the host must find the gateway on the network and establish a TCP connection to it. The WH standard does not define which protocol to use to connect the gateway to the access point, so the authors chose to establish a serial connection with a data rate of 115200 bps. The second part is called NAP-RCP which contains the physical part of the access point and also the firmware. The authors used the Freescale MC1322x platform, which has a transceiver for the complete 2.4GHz ISM radio frequency with low energy consumption. The WH stacks was implemented in the firmware. In order to set up a simple network structure for testing, the authors proposed a virtual gateway, implemented in script format, to forward messages between the network manager and the gateway.

In Rech (2012) the implementation of a network manager following the WH standard was discussed. The work covered the development of the network manager and gateway code that in Hahn (2011) scripts were used for testing. Both the gateway and the NM were implemented as individual processes on a Linux machine. The access point used was the same developed in Hahn (2011) and the NAP-Host is responsible for communicating with the other two processes in the Linux system (gateway and network manager). In this way, three different processes are executed at the main station: NAP, Gateway and Network Manager. The three entities were implemented in C++ programming language, structured in a modular way and object-oriented. The author concluded that the objective of implementing a WH network manager was achieved. The final software is modular and fully expandable according to the needs of the application. Despite this, the work still had several flaws, such as the lack of synchronization of the devices for example, which prevented the author from further testing the stability of the NM. In addition, some stages of the network construction process, such as the service request by field devices, were not addressed by the author.

In the work of Skaar (2012) the general objective is to implement the minimum requirements for the operation of a WH network. The work is a continuation of previous work and proposed the implementation of a NM. To validate the elements developed, tests are presented that include time measurements of communication between several nodes.

3. METHODOLOGY

This section presents some concepts and processes involved to the network manager and also the approaches used for the development of the manager in each aspect.

3.1 Network Manager

Application responsible for creating and maintaining the IWN and its devices. The NM has a direct connection to the gateway, through which it communicates with the applications and the NAP Host, as shown in Figure 2.

The NM has functions responsible for provisioning devices and constantly monitoring the status of the network,

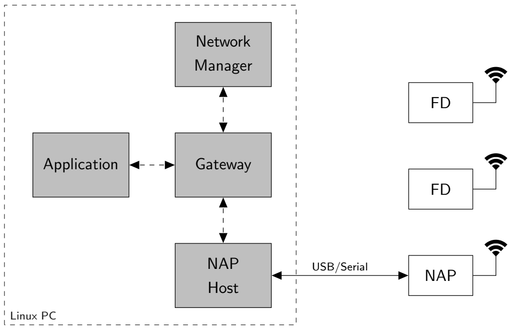

Fig. 2. Test topology.

for example. Among these functions, some of the most important are presented below:

- Device aggregation: The process of aggregating devices on the network (Join Process), is a set of operations orchestrated by the network manager, which ensures the new device aggregation to an existing network occurs properly from the point of view of scheduling, routing and security. Once started, the aggregation process can present the possible results: success, failure, or even be aborted before completion. Despite being orchestrated by the network manager, the aggregation process is initiated through a request from the device that wants to enter the network. The initial steps to perform the device aggregation on the network are: device synchronization with the network, capture of network advertising packets, network aggregation request and security keys transmission. After this initial step, the provisioning takes place, where the device receives several configuration commands from the network manager. It includes writing superframes, links, graphs, device timers setting, among other operations. After the provisioning is completed, the device goes through a quarantine period, until the session is created with the network gateway, thus becoming operational, and starting to perform its tasks. At this stage in the process, devices that wish to periodically publish data on the network must request the service from the NM, and thereby receive the necessary bandwidth, see HART Communication Foundation (2009).
- Scheduling: The process responsible for defining superframes and links that should be allocated to a network device. A superframe is formed of a set of timeslots of 10 ms. The communications scheduling process is performed during the device provisioning on the network, but not necessarily only at this time. In a formed network, there may be a need to reschedule communications, due to changes in the network topology such as the entry of new devices and communication failures between devices. The scheduling process enables that each device on the network knows the communication slots where it must be active, either as a receiver or as a transmitter. The scheduling process is not simple, as it must take into consideration issues such as the publishing rate of each device, transmission priority, redundancy links provision, among other features, see HART Communication Foundation (2008). The links can be organized in different superframes, that means, the links related to the aggregation process and the exchange of management messages between the gateway and the field devices can be allocated in the management superframe while the links used for transmission of the process variables can be allocated on the data superframe.
- Routing: It is the process of creating and maintaining communication routes between network devices. The network manager has complete knowledge of the network over time, and after executing the routing process, it performs the routes distribution to each of the network devices. In this way, the network manager has the complete routing table, and the other devices have subsets of this complete table, only with their routes of interest. As input parameters for the routing table creation, the network manager uses information from the list of neighbors published by each device, in addition to other network statistics, see Chen et al. (2010).
- Network diagnostics: Process responsible for obtaining information about the state of the network. The network devices periodically send information to the manager in relation to the connected neighbors (with link), detected neighbors (without link), in addition to information from the device itself, such as number of received and transmitted packets, power status and package reception errors.

In addition to the functions presented previously, the network manager is also responsible for several other specific tasks related to network security, maintenance, among others. In the case of the network implemented during the development of this work, the network manager is an application that runs on a computer that connects to the gateway through a socket.

3.2 Gateway

The gateway is responsible for making connections between different devices on the network. It is the device that intermediates the data exchange between the manager, the host access point and the automation plant, as seen previously, in the structure of a typical IWN.

The communication between gateway and automation plant can use different protocols and interfaces. This communication type can be used for several functions such as: communications related to process and event data (usually with a predefined publication period), communication of failures and unusual conditions (sporadic communication with critical priority execution), network configuration and maintenance commands (usually infrequently), see HART Communication Foundation (2009). Through this connection, applications can access network devices appropriately. The gateway can also be used as a protocol converter, in a connection between different networks. It is also observed that more than one access point can be connected to the gateway, thus obtaining greater network reliability and enabling greater data flow.

Another important gateway function is clock distribution, which is the primary clock source for the network. The access point connected directly to the gateway is responsible for propagating the clock to the other devices, and in the case of multiple access points, it is the gateway responsibility to synchronize their clocks. In the case of IWN presented in this work, the gateway is an application that runs on the PC, and communicates with other applications through sockets created with the libchan library.

3.3 Access point

Access points are the devices responsible for connecting the wireless devices present on the plant with the gateway. It is possible to have one or more access points, as already shown previously. In the case of the network topology used in this work, there is only one access point, and it connects to the gateway through a host application. The host access point is the interface between the physical access point and the gateway. Communication between the host access point and the physical access point is carried out via a serial bus. The host application accesses a Linux `\dev\ttyUSBx` port to communicate with the physical access point. After network initialization, and before the entry of any field device, the access point is the only device responsible for the propagation of network advertise packets, so that new devices can perform the aggregation process, as presented in the subsection 3.1. The radio used as an access point is shown in Section 4. The set of physical access point and host access point is the interface between the data that travels on the wireless network with the gateway.

3.4 Field Devices

The devices present on the network that connect directly to the plant, and that can have a sensor or actuator function. These devices can be powered by battery, energy harvest or directly in the power grid. In the case of this work, the field devices are not connected to a plant, and the variables that travel on the network are emulated in software. The radio used as hardware for field devices is the same as that used in the access point function, whose characteristics are presented in the section 4.

4. HARDWARE

Devices developed by Müller et al. (2010) were used to carry out the practical tests, with an example shown in Figure 3. The radio used was originally developed in order to meet the requirements for a WH field device, but it also has necessary characteristics to be used as an access point as well.

Fig. 3. *Wireless*HART radio.

The field device has SoC *Freescale* MC1322x as its main component. The MC1322x has a 2.4 GHz radio frequency transceiver and an ARM7 32-bit MCU. This component features hardware acceleration for IEEE 802.15.4 MAC and AES encryption, in addition to several other peripherals, see NXP (2013). The device is used both as field device function and access point in the network used for tests shown in Figure 2.

5. RESULTS AND DISCUSSION

5.1 Adaptive Channel Mapping

The WH standard does not provide the use of adaptive channel mapping, therefore, for channel map adjustment, the network operator need to decide which channels should be removed, and this decision must be based on some prior information in regarding the frequency spectrum occupation. These data can come from measurements performed with auxiliary equipment, spectrum analyzer, for example, or even from preliminary information, given the knowledge of the networks that coexist in the place. In WH commercial devices the option of selecting channels to be used in communications is observed, however this selection is only applied at network startup, that is, in a already operational network, a restart is required to apply the changes EMERSON (2013). As a way of automating the channel mapping process, the work presented in Feldman (2020) presents the development of an adaptive channel mapping technique, where there is no need for prior knowledge of the networks characteristics that are coexisting. In addition, it has an adaptive characteristic, that is, the ability to adjust itself as changes occur in the environment. For this, the topology used is shown in Figure 4.

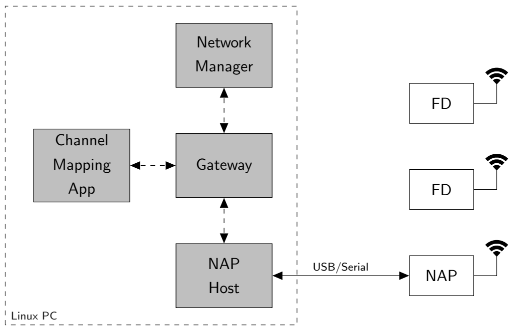

Fig. 4. Channel Map Setup.

The channel mapping system is based on three main stages, which are: Spectrum sensing; Channel map creation; and Channel map update. In addition to the use of the manager developed in this work, for the implementation of this system it was necessary to change the access point and field devices firmware.

The spectrum sensing stage aims to obtain information about the occupation level of each channel, and for this purpose, the energy detection method was used, which in this case is performed at times of idle network Winter and Pereira (2014). The application is responsible to request the sensing obtained values by each of the devices periodically, and thus performing the creation of an experiment matrix that will be used in the next step for channel selecting.

The channel map creation stage aims to use the information obtained in the first stage (spectrum sensing) to create the new channel map to be used in the network devices. Channel selection can be done by different methods. The channel selection methods use the experiments matrix obtained in the previous stage as the main parameter, in addition to specific parameters to each algorithm. The *k-worst* algorithm uses the value k, whereas the selection method with ANOVA uses the confidence level.

The last step of the adaptive channel mapping system is update the channel map in all devices present in the network. The channel map update on the devices cannot be performed instantly on each of them, due to the possibility of channels inconsistency, which would lead to communication failures between devices with different maps. As a way to carry out the update safely, an update scheduling command was developed.

Changes in the field devices and access point firmware were made. It was necessary to update the TDMA state machine to implement the proposed spectrum sensing, in addition to the inclusion of commands for reading the detected energy and also for writing the new channel map.

As a form of example, the work presented in Feldman (2020) brings as one of its results a reduction of 24.82% in the average uplink transmission failure rate with the application of the adaptive channel system as shown in Figure 5. The case study that led to the result presented above was carried out in an environment where a *Wireless*HART network and a WiFi (802.11b) network coexisted.

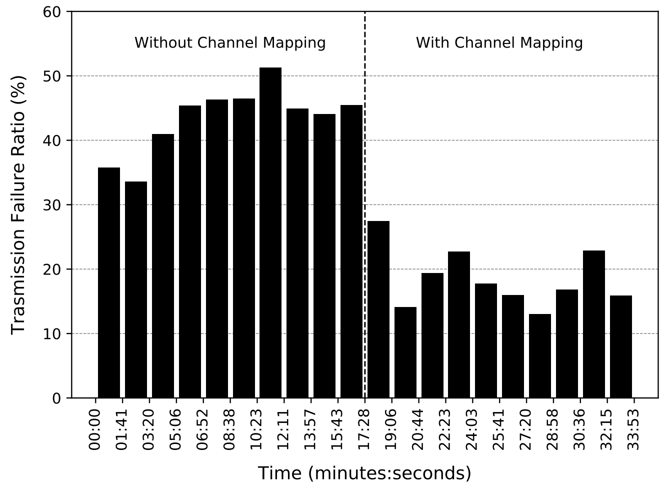

Fig. 5. Transmission fails.

5.2 Link scheduling

There are two possible strategies for organizing data links: (i) a single data superframe for all devices, where each device receives an amount of links proportional to its publication rate; (ii) a data superframe for each publication rate present on the network, with all devices receiving the same number of links but with different superframes. The WH standard does not determine which technique should be used, but it does indicate that the use of multiple superframes can optimize network resources.

Several works propose scheduling algorithms for WH networks, however practical tests are not common, since it is not possible to change the algorithms of commercial equipment. Thus, most authors present simulated results of their work.

The work carried out in Cainelli (2020) proposes a scheduling algorithm that uses the technique of multiple data superframes, that is, each publication period on the network has a specific superframe. The main objective of the proposed algorithm is to reduce the time needed to perform the scheduling process in a centralized manner and using the multiple superframes strategy. In addition to the simulated comparison with other algorithms, a practical comparison was made between the scheduling process generated by the developed network manager that uses multiple superframes (technique ii) and the scheduling process generated by a commercial gateway using a single superframe (technique i). The comparison was performed considering only the data superframes.

For this comparison, a network was formed with three field devices configured with the keys and the network ID of the commercial gateway. Then the same test was performed with the developed NM. These devices have individual identifications and each has a different publication period. Table 1 shows the devices used.

Table 1. Devices used in the test with the developed NM and the commercial manager.

Device ID	Publication period (s)
1002	2
1004	4
1008	8

After the aggregation of each device, when the network was already formed, the command 784 was sent to each devices in order to read the links. The links considered in the analysis are related to data superframes, that is, it was not considered join, discovery or normal links allocated in the management superframe. Through Figure 6 it is possible to graphically visualize the differences in the number of data links that each device receives in the two compared methods.

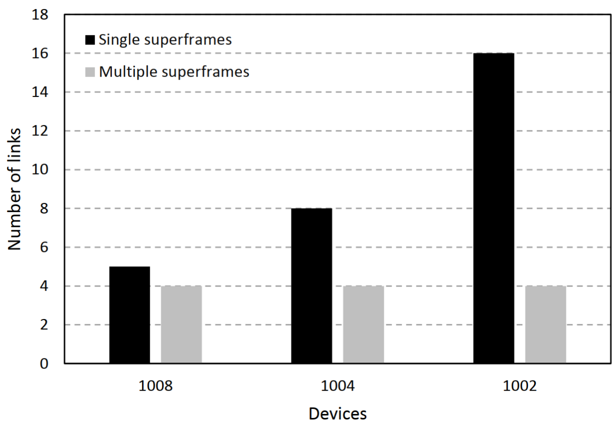

Fig. 6. Number of links received by devices using a single superframe and using multiple superframes.

The commercial gateway uses the technique of a single superframe, and configures this superframe with 1024 timelsots. As timeslots have 10ms, the superframe is repeated every 10.24 seconds approximately. Due to design and robustness issues, the minimum number of data links that the commercial gateway writes to the devices is four, all of which are allocated on the same superframe. Thus, devices that have a publication period greater than 10s receive four links, and devices that publish in periods less than 10s receive more than four links as needed, so during the experiment with commercial equipment, it was observed that each device received a different amount of links.

As for the developed manager, it was observed that all devices received the same number of links. This is due to the fact that the developed NM organizes communications considering that each publication period has a different superframe. In this way the device that publishes its data every 2 s receives 4 links distributed in a superframe of 200 timeslots size, whereas the device that publishes every 4 s receives 4 links distributed in a superframe of 400 timeslots, and so on.

The main advantages of using multiple superframes are greater efficiency in the use of resources (bandwidth savings), reduced memory requirements, reduced network traffic during provisioning and maintenance and energy savings.

5.3 Fast data collection

This application aims to present an alternative technique to collect data from field devices in a faster way. This is suitable for mobile devices that have time constraints to operate, which is the case, for example, of devices that intermittently need access to the network. Each time a new device needs to connect to a WH network it is necessary to perform the join sequence, then the device is ready to fulfill its duties in the network. The join sequence is a series of commands exchange (in request and response form) between the new device and the NM. Due to centralized control characteristics of the WH networks and the mesh network topology, this join process is associated with large times given that the communications may take several hops to reach its destiny. Considering a moving device, it may not have enough time to complete this process before leaving the network coverage area.

The main steps of the standard join sequence are: periodic advertise packets by network members (these packets allow the network to be identified); monitoring by the new device to locate and synchronize with the network; establishing a secure channel between the new device and the NM; verifying the trustworthiness of the new device; provisioning the device and allow it to integrate the network. All these steps in addiction with other tasks, such as retransmission, are necessary to maintain the security and reliability of the standard. In the other hand, it precludes the use of mobile devices because of the time needed to complete the process.

This case study proposes a modified join sequence in order to offer support for mobile devices to fulfill its duties in the network, which is in this case sharing its dynamic variables at the gateway. Initially, the field device was modified to generate a different join request including already the dynamic variable. In order to identify the modified join request, some changes were made in the network manager and in the gateway. In this case, the gateway is able to process two different join requests. Each time a join request reaches the gateways, it recognizes as a message coming from a new device and request to the NM a join session for the new device. Then, the message is deciphered and the commands are extracted and identified as standard or modified join request. From this point on, each case operates differently. In the modified situation the dynamic variable is stored in a gateway buffer and the join process is ceased. Thus, the number of communications is reduced and the time to perform the task too.

This application explores the capability of the developed network manager to deal with situations not covered by the standard allowing the practical analysis of proposed techniques to improve the operation and performance of industrial wireless networks.

6. CONCLUSION

This work presents the development of a network manager compatible with the *Wireless*HART protocol, so the use of commercial field devices is possible, with no difference in relation to a commercial manager. Despite the compatibility with the mentioned protocol, the use of this manager also allows the inclusion of new techniques and protocol modification, enabling new use possibilities. A structure consisting of an access point, gateway and network manager was developed, in addition to the presence of an interface for the use of auxiliary applications.

The use of the developed network manager was carried out during the case studies execution in different works, such as the scheduling algorithms study, adaptive channel mapping and the development of fast data collection in industrial wireless networks. In all these works, the use of this network manager was essential, as it allowed the necessary customizations for each of the applications, which would not be possible in many cases with the use of a commercial device.

Among some future works that will use this manager for the network implementation, we can mention the study of routing algorithms, automated exchange of neighbors in a network and the networked control integration with network control.

REFERENCES

Cainelli, G.P. (2020). *Proposta de algoritmo de escalonamento de mensagens para redes sem fio aplicadas à automação de fábrica.* Dissertação (mestrado em engenharia elétrica), Universidade Federal do Rio Grande do Sul, Porto Alegre.

Chen, D., Nixon, M., and Mok, A. (2010). *WirelessHART - real-time mesh network for industrial automation.* Springer.

EMERSON (2013). Emerson wireless 1420 gateway. Available at: <https://www.emerson.com/documents/automation/folha-de-dados-do-produto-gateway-smart-wireless-emerson-pt-87348.pdf>. Accessed: 31/01/2020.

Feldman, M. (2020). *Técnica de Mapeamento Adaptativo de Canais para Redes Sem Fio Industriais.* Dissertação (mestrado em engenharia elétrica), Universidade Federal do Rio Grande do Sul, Porto Alegre.

Hahn, D.H. (2011). *Desenvolvimento de um ponto de acesso para redes WirelessHart*. Projeto de diplomação (graduação em engenharia elétrica), Universidade Federal do Rio Grande do Sul, Porto Alegre.

HART Communication Foundation (2008). HCF_SPEC-075 - TDMA data link layer specification.

HART Communication Foundation (2009). HCF_SPEC-085 - Network management specification.

Müller, I., Pereira, C.E., Netto, J.C., Fabris, E.E., and Allgayer, R. (2010). Development of a wirelesshart compatible field device. In *2010 IEEE Instrumentation Measurement Technology Conference Proceedings*, 1430–1434. doi:10.1109/IMTC.2010.5488129.

NXP (2013). MC1322x datasheet. Available at: <https://www.nxp.com/docs/en/data-sheet/MC1322x.pdf>. Accessed: 31/01/2020.

Rech, J.R. (2012). *Desenvolvimento de um gerente de rede WirelessHART*. Projeto de diplomação (graduação em engenharia elétrica), Universidade Federal do Rio Grande do Sul, Porto Alegre.

Sanchez, J.H. (2011). *WirelessHART network manager*. Master's thesis, KTH Royal Institute of Technology.

Skaar, A.A.R.V.S.K.F.L. (2012). Design and implementation of a rudimentary wirelesshart network. Technical report, University of Oslo - Department of Informatics.

Winter, J.M. and Pereira, C.E. (2014). Coexistence aware for wirelesshart networks. In *2014 12th IEEE International Conference on Industrial Informatics (INDIN)*, 803–806. doi:10.1109/INDIN.2014.6945618.