



HAL
open science

Media independent solution for mobility management in heterogeneous LPWAN technologies

Wael Ayoub, Abed Ellatif Samhat, Fabienne Nouvel, Mohamad Mroue,
Hassan Jradi, Jean-Christophe Prévotet

► To cite this version:

Wael Ayoub, Abed Ellatif Samhat, Fabienne Nouvel, Mohamad Mroue, Hassan Jradi, et al.. Media independent solution for mobility management in heterogeneous LPWAN technologies. *Computer Networks*, 2020, 182, pp.107423. 10.1016/j.comnet.2020.107423 . hal-02938593

HAL Id: hal-02938593

<https://hal.science/hal-02938593>

Submitted on 26 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Media Independent Solution for Mobility Management in Heterogeneous LPWAN Technologies

Wael Ayoub, Abed Ellatif Samhat, Fabienne Nouvel, Mohamad Mroue, Jean-christophe Prévotet and Hassan Jrade

ARTICLE INFO

Keywords:
LPWAN
Handover
Mobility Management
Seamless Connectivity
Heterogeneous Technologies

ABSTRACT

The integration of IoT communication technologies such as Low Power Wide Area Networks (LPWAN) is essential in today systems including supply chain, healthcare, autonomous vehicle, etc. In such environments there is an increase in IoT applications that require End Devices (EDs) mobility within similar or heterogeneous LPWAN technologies. Supporting mobility refers to ensure the delivery of data on demand and during movement. In this paper, we investigate the mobility of ED between different LPWAN technologies. We propose a new mobility management solution in LPWAN to achieve continuity of the communication after the link layer technology changes between the ED and the Application Server (AS). This solution is IPv6-based and supports switching between heterogeneous LPWAN networks and technologies. It results in reducing the time required to deliver the data by optimizing the communication routing and saving the bandwidth. The validation of the proposed solution is achieved through the scenarios of ED handovers between LoRaWAN and NB-IoT using simulations as well as experimentation.

1. INTRODUCTION

The Internet of Things (IoT) is a collection of interconnected physical devices that can monitor, report, send and exchange data. Among IoT technologies, LPWAN combines the wide range and the low power consumption, which makes it powerful for a variety of IoT use cases. LPWAN fills the gap between mobile communications such as 4G and LTE, and short-range wireless communications such as WiFi and Bluetooth. In LPWAN, EDs typically connect to AS through various technologies. EDs may consist of smart devices or mobile sensors. EDs use sensors to measure specific aspects of the surrounding environment including location, temperature, humidity, light levels, movement speed and others. For example in the supply chain, EDs are an effective way to track and authenticate products and shipments using GPS and other technologies. They can also control/monitor the storage conditions of products, which improves quality management throughout the supply chain. In the supply chain, trucks or other means of transports can move within a city, between different cities, or between countries.

But the challenges of using IoT in the supply chain or other systems is based on admitting good network connectivity for EDs to work well. They must be able to communicate their positions during mobility. Thus, mobility in this context is defined as the act of moving between Gateways (GW) of different operators and different technologies as shown in fig. 1. Therefore, to achieve timely delivery of information, ED may support more than one technology.

Before LPWAN, tracking and monitoring products in supply chain were done using the conventional cellular network

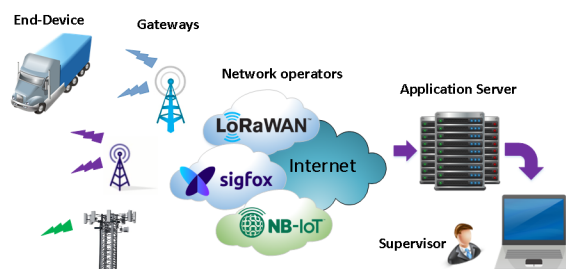


Figure 1: Tracking Location

[1]. Cellular network is widely implemented and it supports roaming of the ED while moving between countries. Actually new technologies are gaining a significant share of the market as LoRaWAN [2], NB-IoT [3], etc. Its implementation is growing exponentially to include more than monitoring static applications as shown in [4]. These technologies can support multi-domain of IoT applications, such as smart infrastructure, transportation, industry, retail, etc. Using of LPWAN instead or as an alternative to cellular network, which is expensive as shown in [4], will reduce the cost. Therefore, investigating the use of LPWAN in IoT applications that require long-range connectivity and mobility as in the supply chain is very interesting.

In a previous study [5], the authors showed that EDs can access the Internet through the Home or Visited network operator using the same technology. This type of mobility is defined as roaming. Roaming was essential for collaboration between network providers to allow ED to move from one network to another using the same link layer technology. In this work, changing the link layer which means changing the technology requires specific mechanisms to support mobility. Therefore, ED can subscribe to an Internet connection using multiple technologies. The objective is to propose a

✉ wael.ayoub@insa-rennes.fr (W. Ayoub); samhat@u1.edu.lb (A.E. Samhat); Fabienne.Nouvel@insa-rennes.fr (F. Nouvel); mohamad.mroue@u1.edu.lb (M. Mroue); jean-christophe.prevotet@insa-rennes.fr (J. Prévotet); hassan.jrade@insa-rennes.fr (H. Jrade)
ORCID(s):

mobility management solution between heterogeneous LPWAN to ensure the continuity of the communication after the change of the L2 and L1 layers of the protocol stack.

In this context, to the best of our knowledge, this problem is mainly addressed with Media Independent Handover (MIH) of the IEEE 802.21 standard [6]. MIH framework requires a lot of messaging between the ED and the core network. Thus, this standard is heavy on constrained IoT applications that have to consider the duty cycle the message rate limitations. This motivated us to propose a light weight solution which is an alternative to MIH and can be generic for EDs to support more than one technology. Furthermore, we benefit from the use of IPv6-based solution [7] to support continuity of the communication after the link layer changes. In this solution, the IPv6 layer will be a common layer between any ED that implements our solution and the IP-based Internet core. Thus, the adaptation of IPv6 with the IoT device, taking into account all its limitations, will support ED with the ability to communicate with any other ED and application independent of the used technology. In addition, for connectivity and mobility, the IPv6 protocol provides the IoT device with many functions and features, as detailed in the previous work [8]. The proposed mobility management solution, between different LPWAN technologies, uses the concept adopted in the standard Static Context Header Compression (SCHC) [9]. The SCHC standard has defined a mechanism to compress / decompress protocol headers in less than 2 bytes to limit the additional overhead on the radio frame. We validate the solution using LoRaWAN and NB-IoT technologies where the ED is registered with both networks.

The rest of this paper is organized as follows, we present the two LPWAN technologies, LoRaWAN and NB-IoT as they will be used as an example scenario to test the proposed solution. In section 2, these technologies are summarized in terms of the connection mechanism and registration process. Then, in section 3, we present our proposed Solution in terms of the network architecture and the protocol stack in the ED followed by the communication procedure and handover steps. In section 4, we specify the steps taken by ED to alternate between the two technologies in three scenarios: Connection lost, Handover initialized by the network and Handover initialized by the ED. In section 5, we show the contributions in terms of supporting switching between heterogeneous networks and technologies, and granting the use of IPv6-based solution which results on saving the time required to deliver the data by optimizing the communication routing and saving the bandwidth. In addition, in section 6, we highlight the cases within the same technology by improving LoRaWAN roaming and decreasing the registration process of NB-IoT ED to access the network. Finally, section 7 provides a conclusion and presents some future works.

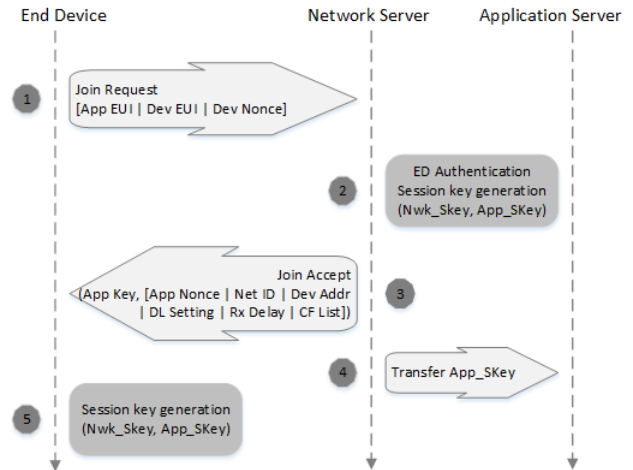


Figure 2: LoRaWAN Registration Process

2. Connection and Registration Processes of the considered technologies

LPWAN is ideal for IoT applications that only need to transmit small amounts of long-range data, such as in the supply chain, etc. As LoRaWAN and NB-IoT are a matter of interest to researchers and industry, we consider them as a case study to validate our proposal. In the following subsections, we provide a general description of the two technologies. Then, we study the exchange of messages and the time required for ED to access the network.

2.1. LoRaWAN

LoRaWAN is an open standard architecture developed by LoRa Alliance [10] to provide a connection to a restricted wireless sensor device. This standard allows the ED to communicate with one or more gateways covering its location. LoRaWAN is a link layer protocol that runs over the LoRa physical layer protocol.

2.2. NB-IoT

The Narrow Band Internet of Things (NB-IoT) is part of version 13 [11] defined by 3GPP. It has been configured into the cellular system to support extremely low complexity and low performance communication. It defines a new radio access technology that can be integrated into the LTE standard. NB-IoT is built from existing LTE features, but it eliminates many features to keep this standard as simple as possible, to reduce the cost of the device and to minimize battery consumption.

2.3. Registration Mechanism

To participate in the network activities, a registration procedure is required for the ED either in LoRaWAN or NB-IoT.

2.3.1. LoRaWAN

In LoRaWAN [10], the specification defines two registration procedures: Activation By Personalization (ABP) and On-The-Air Activation (OTAA). In the ABP, communication is not required to join the network since ED has the

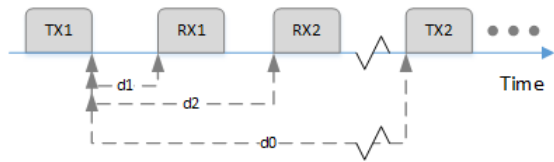


Figure 3: Medium access in LoRaWAN

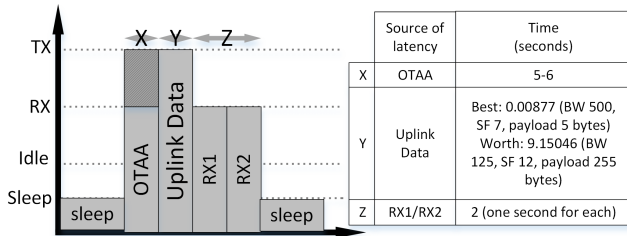


Figure 4: Medium access and latency source in LoRaWAN

required parameters and headers in the memory. In OTAA, as shown in fig. 2, the ED follows a procedure to obtain the required keys to join the network. It starts by sending a join request to the GW in TX1, then the GW forwards the frame to the network server (NS). Then, NS responds with a join accept sent by the gateway. A first reception window is opened after $d1 = 5$ seconds from TX1 as shown in fig. 3. The gateway can transmit this join accept, either during the first RX1 window or during the second RX2 reception window. If no preamble is detected during RX1, the second RX2 reception window will open after $d2 = 6$ seconds from TX1. Whether the preamble is received or not, the channel cannot be used for a period of time $d0$, depending on the duty cycle. After the OTAA process, ED can send an uplink frame when the data is available for transmission. After the activation process and in normal transmission, RX1 will open after $d1 = 1$ second from TX1 and RX2 will open after $d2 = 2$ seconds from TX1. Each of the RX1 and RX2 opens for one second. As shown in fig. 4, the uplink process needs between 8.77 ms in the best case and 9.15 seconds in the worst case. In fact, we have to take this time delay into account when processing the handover. We can see that the time to register on the network is quite long and it is not done immediately.

2.3.2. NB-IoT

In NB-IoT [11], ED searches for a GW at an appropriate frequency. Then, the registration setup begins, as shown in fig. 5. During the setup, the ED starts listening to obtain the Narrow Channel Physical Cell (NCellID) identifier from the Narrow Band Synchronization Signals (NPSS/NSSS) channel transmitted by the GW. Then, the ED decodes NCellID to obtain the Master Information Block (NB-MIB), which includes the size of System Information Blocks (NB-SIB1), the number of repetitions, the InfoSIB1 programming (access and cell selection) and its starting position. Then, the ED decodes SIB1-NB to obtain cell access parameter information: cell identity and status, and cell selection informa-

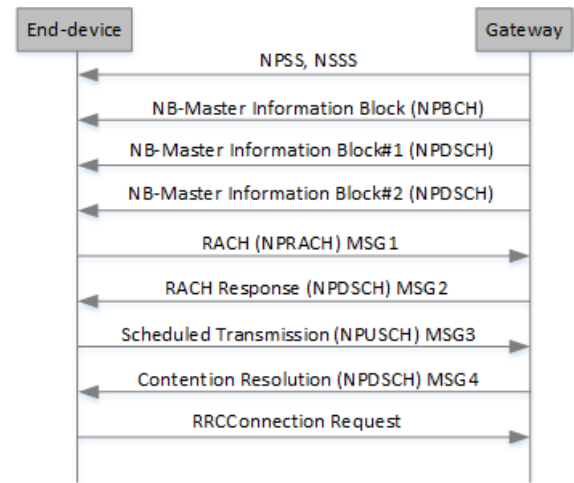


Figure 5: NB-IoT Registration Process

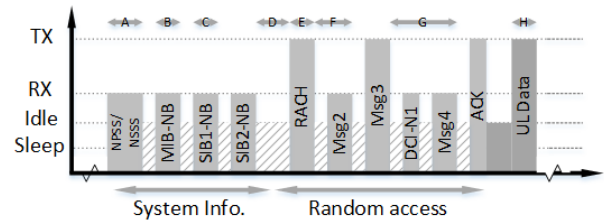


Figure 6: Medium access in NB-IoT

tion, such as the minimum level of the cell, etc. Then, the ED decodes the NB-SIB2, which provides configuration information for common logical and physical channels. Most of the information in SIB2 is about the Random Access Channel (RACH) configuration required for uplink synchronization. At this level, the ED initializes and sends the RACH preamble to the GW in message 1. The ED will repeats until a GW response. When the GW receives the request, it responds with message 2. Then, the ED sends the message 3 to start the content resolution process and GW sends the response in the message 4 indicating that the RACH procedure has been completed. Finally, RRCConnectionRequest states that the ED wants to connect to the network.

As shown in fig. 6, the procedure for accessing the NB-IoT ED medium begins upon receiving synchronization and acquiring system information [12]. After decoding MIB-NB and at least SIB1-NB and SIB2-NB, the ED obtains the GW access configuration. As shown in [13] and reported in Table 1, the time required to obtain system information varies between 24 ms and 2604 ms. In addition, in [14], the authors distinguish between the deployment of NB-IoT as a stand-alone or in-band and as a guard band. The results of the tests show that in the stand-alone case the time required for synchronization and MIB acquisition is equal to 1160 ms, while for the in-band case it is 3030 ms.

In summary, the registration process, including synchronization and acquisition of system information, in NB-IoT technology requires a time delay between 8 and 10 seconds

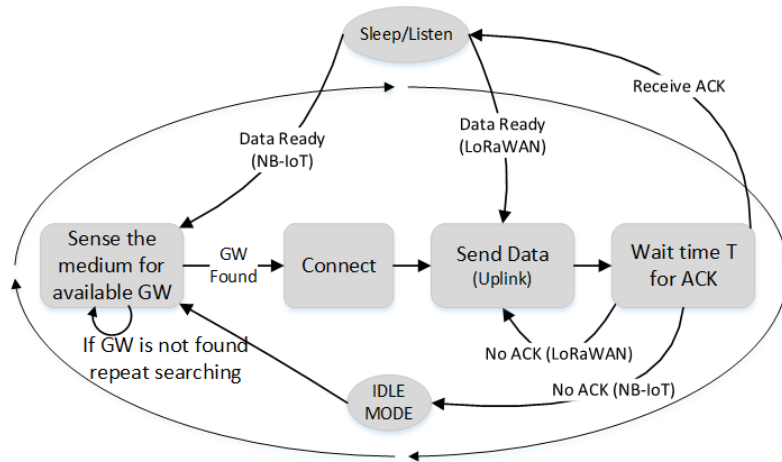


Figure 7: Connection and exchange mechanisms scheme for ED in LoRaWAN and NB-IoT

Table 1

Latency source in NB-IoT

| | Source of latency | Time |
|-------|--|--|
| A | Initial synchronization | 24 ms / 2604 ms |
| B & C | System information | MIB: 640 ms SIB1: 2560 ms SIB2: chosen by operator |
| D | Narrowband Physical Random Access Channel (NPRACH) | 40, 80, 160, 240, 320, 640, 1280, 2560 ms |
| E | Preamble transmission | Best: 5.6 ms (Format 0, 1 repetition) Worst: 819.2 ms (Format 1, 128 repetition) |
| F | RAR reception | $256 \times (0, 1, 2, \dots, 2048)$ ms Worst: ~9 minutes |
| G | Contention resolution window | Best: 4 ms Worst: 2.3 hours |
| H | Narrowband Physical Uplink Shared Channel (NPUSCH) | Best: 1 ms (15 kHz, multi-tone, shortest RU, 1 repetition, 1 RU) Worst: 40,960 ms (3.75 kHz, 128 repetitions, 10 RUs) |

[15, 16]. Whereas, it is between 5 and 6 seconds for LoRaWAN [5, 4]. These values will also represent the time required by ED to access each network after handover.

2.4. Connection and exchange mechanisms

Fig. 7 illustrates the communication process of the ED to connect and to transmit / receive. In LoRaWAN, ED is initially in sleep or in listen mode. The ED of this technology does not have a verification process before transmitting. The LoRaWAN ED transmits the data as soon as they are ready. Then, it listens on RX1 window for the acknowledgment (ACK). If ACK is not received, the ED re-sends the

data. This process is repeated according to the number of repetitions configured in the ED. This number can be fixed or managed according to the priority of the data. When ACK is received, the ED goes back to sleep mode. In our framework we use ACK to confirm that data delivered during mobility.

For NB-IoT ED, each device is associated with a GW. As soon as the data is ready to send, the ED will be activated. Once established, the connection is resumed. Otherwise, the ED will search for a GW to connect. As the connection is established, the ED transmits the data and then waits for the ACK. In case the ACK is received successfully, ED returns to sleeping as shown in fig. 7. Otherwise, ED returns to IDLE_mode to sense the medium for available GW on the same frequency. In Idle mode, the ED radio is still on and listens for nearby incoming GW signals. After processing the incoming messages, the ED detects whether the nearby GW is the one that is registered to or not. Then, ED resumes the connection. In the second case, the ED must register with the new GW before repeats the data transmission. This process is repeated according to the priority of the data and the number of repetitions configured in the ED.

After having presented the two technologies, we will propose in the following section a framework which addresses the limits and restrictions imposed on ED to establish a connection and transmit data. In addition, the framework is dynamically adapted with ED and the various architects of the networks.

3. proposed solution

3.1. Protocol Stack

Usually, EDs implementing IP-based protocol stack support: an application layer protocol such as CoAP, followed by a transport layer protocol such as UDP and then a light IPv6 based protocol in the network layer [17, 18]. We propose a stack composed of three levels, as shown in fig. 8, to manage the handover from one technology to another.

3.1.1. Upper-Level

This level has to manage the mobility and select one of the technologies found on L2 layer (data link layer). This level contains three blocks:

- **Protocol stack:** it is formed of the protocols UDP and light implementation of Mobile IPv6 (LMIPv6). Other than these three protocols shown in fig. 8 can be used while considering the ED limitations in term of hardware resources and restrictions of the radio. The protocol stack block is responsible for the management of mobility using LMIPv6 and the continuity of the session with AS after the handover between networks.
- **Technology Selection Algorithm (TSA):** it is responsible of the selection of the technology as detailed in [19].
- **Database:** it contains the context for compression and decompression (C/D) of the protocol stack headers. Moreover, it contains the parameters to reconfigure the dynamic layer at the middle-level.

3.1.2. Middle-Level

Called Adaptation Layer for Connectivity (ALC). At this level, we define three services and the Dynamic Context Header Compression (DCHC) block.

- **ALC Event Service (ALES):** it provides the upper layers with the classification, filtering and notification of events corresponding to dynamic changes in the characteristics of the link. When a node is enabled, the protocols in the stack block must be registered with the services provided by the ALES system to manage the data flow and decide whether to perform the handover procedure or schedule it.
- **ALC Command Service (ALCS):** it allows the ALC layer to manage and control the behavior of the links to plan the handover procedures.
- **ALC Information Service (ALIS):** it defines a framework for the acquisition, storage, and retrieval of information relevant to the transfer decision.
- **DCHC block:** it is responsible for the C/D and fragmentation of data into L2 frame. This block is configured by the ALC [20] and will be illustrated in the following sub-section B.

3.1.3. Lower-Level

This level includes more than one technology. It provides events and triggers about the data link layer using ALES to middle-level. In addition, it uses the ALIS to send the data and information to upper-level passing through the middle-level. Then, it receives the configurations and comments using ALCS to select the technology.

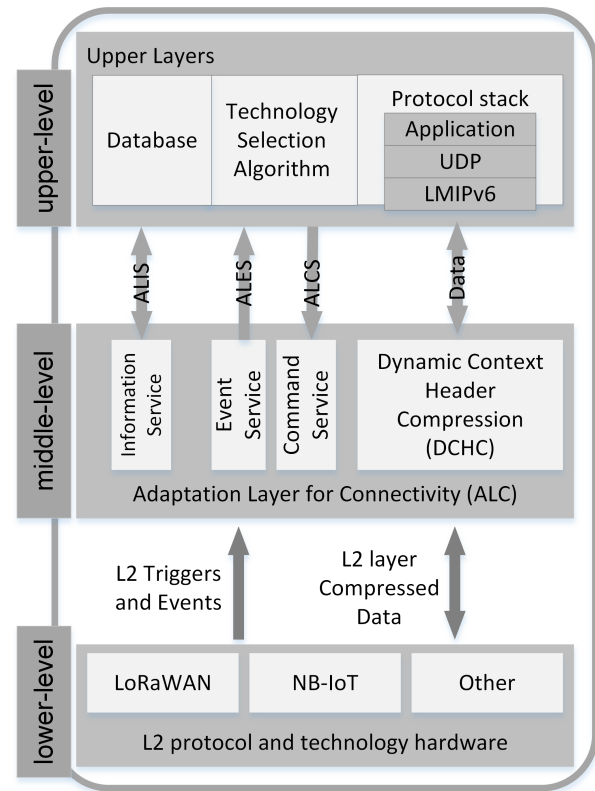


Figure 8: ED Protocol Stack

3.2. Dynamic Context Header Compression (DCHC)

By default, the Static Context Header Compression (SCHC) [21] standard was developed to seamlessly connect the protocol stack at upper-level with one of the lower-level technologies. SCHC is responsible for the C/D of protocol headers found in the protocol stack. Moreover, SCHC fragments the final packet before transmitting it. This is done in case the packet did not fit within the frame of the used technology. However, with all of its features, SCHC only considers static headers. While, during mobility, the values of packet headers will change, such as the source IP, the destination IP, etc. Until now, the SCHC protocol is statically configured for a single technology. However, the standard was developed and updated to support many IoT technologies such as LoRaWAN, Sigfox, NB-IoT, etc. Therefore, this motivated us to create a dynamic layer to reconfigure SCHC during runtime when the L2 layer technology changes. This work considers that ED supports more than one technology.

The DCHC mechanism is a header compression scheme, inherited from the SCHC standard. DCHC compresses the protocol headers in the protocol stack into less than 2 bytes. The DCHC compression mechanism takes into account the dynamic change of headers during mobility. DCHC is placed at the middle-level to operate between the network and the different data link layers supported by ED.

As in SCHC, the DCHC mechanism compresses UDP / IPv6 headers into RuleID and sends the data to the lower

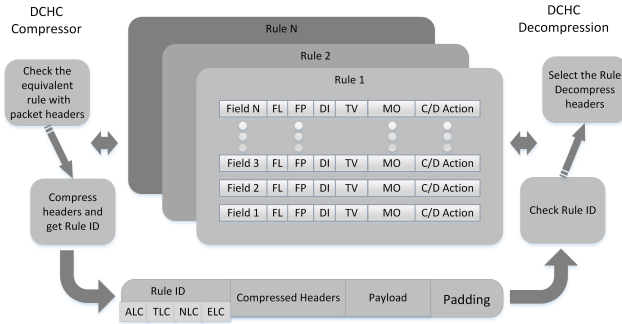


Figure 9: DCHC Mechanism Implementation

layers for transmission as shown in fig. 9. DCHC RuleID is formed by the concatenations of the RuleIDs of each layer of the protocol stack as shown in fig. 10. DCHC proposal consists of several contexts, rather than the unique context for the SCHC. Each context contains rules for a single layer. In DCHC, the RuleID has been divided into segments; each segment represents a layer as shown in fig. 9. DCHC deals with context by layers instead of SCHC. As shown in fig. 10, the application layer context is used to compress the fields of the used application layer protocol. The same concept applies to transport, and network layers. The context is saved in ED and the operator's network server consists of rules. Each rule consists of several fields. Each field corresponds to a parameter from the header. A rule holds the information of the header. Each rule is represented by a RuleID that represents the compressed header. The RuleID is formed by:

- Field ID (FID): A unique value used to represent the header field.
- Field Length (FL): Used to specify the header segment length.
- Field Position (FP): In case the field is an array, field position is used to select a specific item in the array.
- Direction Indicator (DI): Used to specify the direction of the packet, whether uplink (Up); data are from device to server, downlink (Dw); data are from the server to device, or Bidirectional (Bi).
- Target Value (TV): This field contains the value that ED receives to compare with the saved value.
- Matching Operator (MO): This operator is applied in comparing received TV with saved one.
- Compression Decompression Action (CDA): It describes the method used beside each rule field to compress and decompress the TV.

The configuration parameters are obtained from the database to configure DCHC when TSA selects a technology. After each configuration, DCHC reforms the data received from the protocol stack block to adapt the payload frame of the technology used at lower-level.

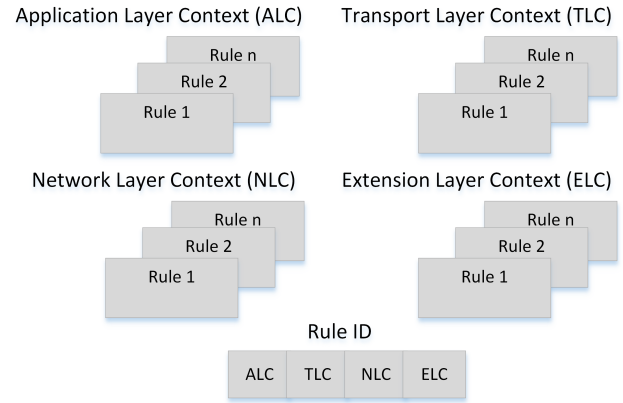


Figure 10: Proposed DCHC RuleID

3.3. Network Architecture

Basically, LPWAN technologies share a similar system architecture but with different terminologies [8]. In this study, we consider the scenario of the supply chain where an ED is installed in a truck that carries food. During the movement, the ED leaves the coverage of the NB-IoT network and enters a zone covered by LoRaWAN technology. The two networks are connected to the Internet core. The AS is responsible for monitoring the different information provided by the ED. As the network and location change, the AS and the truck stays the same, but the network operator and the GWs change. Thus, we propose to create a dynamic solution that can address the heterogeneity of the networks to contribute in communication as a network operator and GWs changes [7]. This solution is called the Mobility Management Server (MMS) which will be responsible for monitoring ED movements and contributing to the management of ED handover. In addition, MMS will manage the registration and exchange of the DCHC context with the new operator. Furthermore, MMS will track the geographic location of the devices and share these records with other operators to avoid the loss of their transmitted messages and to decrease the latency in the delivery of the data to AS as shown in [5, 7]. Therefore, MMS includes a geolocation table for EDs that belong to the network. Also, it periodically updates the table by connecting to the central server of each technology. But, the challenge is in the MMS implementation strategy. Therefore, we propose to integrate MMS in three different ways to address mobility in different types of networks. To allow an efficient implementation of MMS to easily exchange data with NS and ED via GW, we propose three different implementation scenarios:

Scenario 1: A single network operator with multiple LPWAN access technologies as shown in fig. 11. In this scenario, MMS is implemented within the core of the network operator.

Scenario 2: Multiple network operators with multiple LPWAN access technologies, as shown in fig. 12. In this scenario, operators must provide the following MMS functionalities to allow seamless handover:

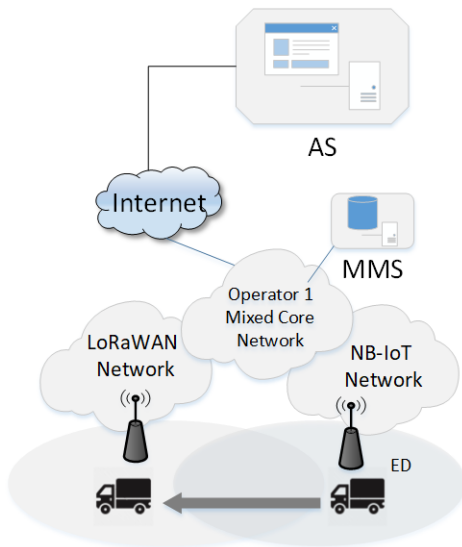


Figure 11: First MMS deployment scenario

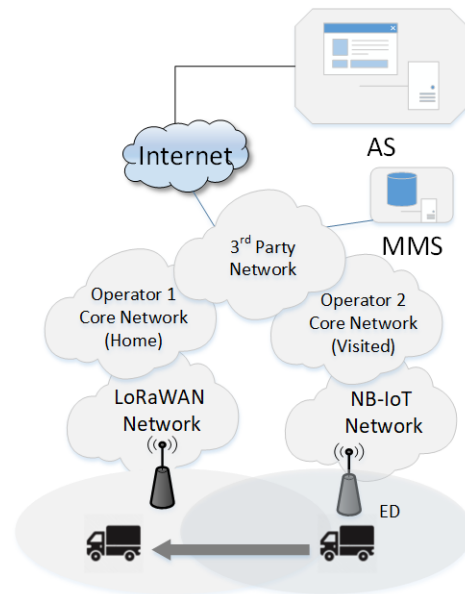


Figure 13: Third MMS deployment scenario

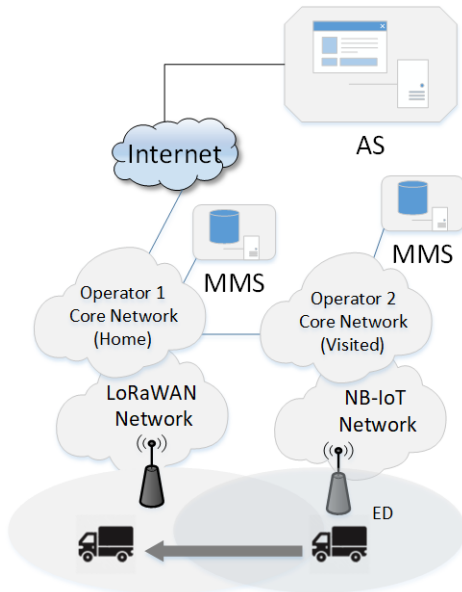


Figure 12: Second MMS deployment scenario

- All network operators should consider the network server to provide ALES and ALCS services to ED.
- Each operator can have its own MMS. If the visited network does not provide information services, the ED can access those services through the network server located in its home network.
- To allow EDs to obtain information services from the MMS network they visit, operators of home and visited networks must collaborate and have agreements regarding that.

Scenario 3: Multiple network operators with multiple LPWAN access technologies, where there is a central third-party network to interconnect the central networks of opera-

tors and provide MMS services as shown in fig. 13. Usually, the third-party network is the main one among them. In this scenario:

- All network operators should consider that Network Servers present the services to EDs.
- The MMS is located in the central third-party network, which allows all EDs to access information services from any access network. This scenario requires a collaboration and data sharing agreement between network service operators and the third-party network.

3.4. Communication and handover process

In this section, we describe the handover that an ED can support in such networks. During the movement and when the data is ready to be sent, the ED follows the communication mechanism shown in fig. 7. The received signal strength (RSS) value can be found in the received message that contains the acknowledgement (ACK). Therefore, as the RSS decreases, ED knows that it is moving away from the GW. In addition, ED can detect that a movement happens when an ACK is not received for a transmission. After that, the ED follows the procedure explained in the section 2 to achieve the Internet connection, as shown in fig. 3 for LoRaWAN or in fig. 5 for NB-IoT.

Consider the case when RSS degrades with the current network, while a better connection with another network can be achieved. In addition, for a device that supports multiple technologies, ED can switch to another technology to save costs or energy consumption.

Before starting with the selecting the type of the handover, we have considered the simplicity of LPWAN plus the limitations that ED suffers from. We take into account radio limitations in terms of bandwidth and message rate, interference when using the same frequencies and lost packets due

to collision, as well as the restriction of energy consumption. To illustrate our idea in a better way, consider the handover between two technologies that use the same band as SigFox and LoRaWAN [22, 4]. Connecting with another technology before loosing the connection with the current one requires two radios to be in transmission / reception at the same time. This leads to platform noise and coexistence issues for radios that work in adjacent or close frequency bands and leads to higher costs of ED in general. Furthermore, the simultaneous use of the two radios causes interference and collisions. This may prevent the delivery procedure. To overcome these problems, it is required that the ED have a single radio that transmits at any time throughout the handover process. In addition, constrained IoT technologies are not developed or used in real-time applications. Thus, a delay of up to 10 seconds is still acceptable. To ratify this, the normal registration process in NB-IoT lasts up to 10 seconds [14] as explained in section 2. In addition, due to the energy use restriction, the ED cannot activate both radios at the same time. This results in draining the battery power. Therefore, we believe that disconnecting from the current network then connecting with the new one is the most appropriate handover for our case study.

In conclusion, the strategy of "disconnect then reconnect" is the most suitable one for an ED that is developed for constrained applications in case of switching between two technologies. The handover is divided into different stages:

- Initiation (search for a new link)
 - Network discovery
 - Network selection
 - Handover negotiation
- Preparation (setup the new link)
 - Lower and upper connectivity
 - Resource reservation
- Execution (Transfer the connection)

The proposed ALC in section 3 provides a media independent framework for the means to support the initiation and preparation of the handover. Whereas, the continuity of the communication and execution is the role of the protocol stack found at the upper layers as illustrated in section 3 and proved in [5]. In the next section, we will study the case of an ED making a handover between LoRaWAN and NB-IoT networks and vice versa. Also, we will draw the communication scenario.

4. Message flow during mobility between two LPWANs

Handover between technologies can occur in the following cases:

- Network initialization handover: MMS detects the degradation in ED performance and informs the ED about the necessity of the handover while ED is connected.

- ED initialization handover: ED detects a degradation in RSS or any other parameters. This drives the ED to search for a better connection with another available technology while it is still connected.
- Connection lost: In this case, the connection is lost and the ED is out of network reach.

4.1. Network initialization

As shown in fig. 14, ED initially begins connecting with LoRaWAN network. When the data is ready to send, the data is packaged in the protocol frames of the protocol stack at the upper-level. Then, the last frame, which is the IP packet, is sent to the middle-level. At the middle-level, the protocol headers of the IP packet will be compressed using the DCHC block. Then, the compressed frame will be encapsulated into the LoRaWAN frame. The LoRaWAN frame will be transmitted and captured by any gateway belonging to the ED network operator. At the core of the network operator, the LoRaWAN network server verifies the identity of the received LoRaWAN frame. After this verification, the network server verifies the corresponding application server (AS). Then, the network server loads the context of the ED to decompress the headers of the compressed frame. Then, the IP packet is sent to the AS.

In parallel, the network server updates the MMS with the new parameters of the received frame, such as RSS. Following the reception of ED IP packet, the AS returns an acknowledgment. This acknowledgment is compressed by the network server and forwarded by the closest GW to the ED as shown in section II of [4]. We use the acknowledgment sent by the AS to confirm the reception of the IP packet, whereas the acknowledgment is sent by the network server to confirm the reception of the LoRaWAN frame. If an error occurs during the communication procedure, ED repeats the transmission.

Now, consider the scenario where the ED is far from the GW. Then, the transmitted LoRaWAN frames will be received with a lower RSS. The network server will update the MMS with the new RSS values. Therefore, MMS will detect a degradation in the ED RSS. Based on the collaboration between MMS and the two networks, LoRaWAN and NB-IoT, MMS will connect to the NB-IoT network to verify if a base station is available close to the current location of the ED. Then, the NB-IoT network verifies and returns the information with the available resources. Upon receiving the information, MMS sends a message to the ED using the LoRaWAN network server. This message includes a command to switch to the NB-IoT network. In addition, this message contains useful parameters such as MIB and SIB resources to improve connectivity with the GW of the NB-IoT network. Once ED receives this message, it forwards the content to the upper layers. In the upper layers, the TSA block decides whether to change the technology now or schedule it. When TSA decides to change, it updates the database with the received parameters and then sends a command using the ALCS to the ALC layer. Then, ALC obtains the new database parameters to reconfigure the DCHC to work with

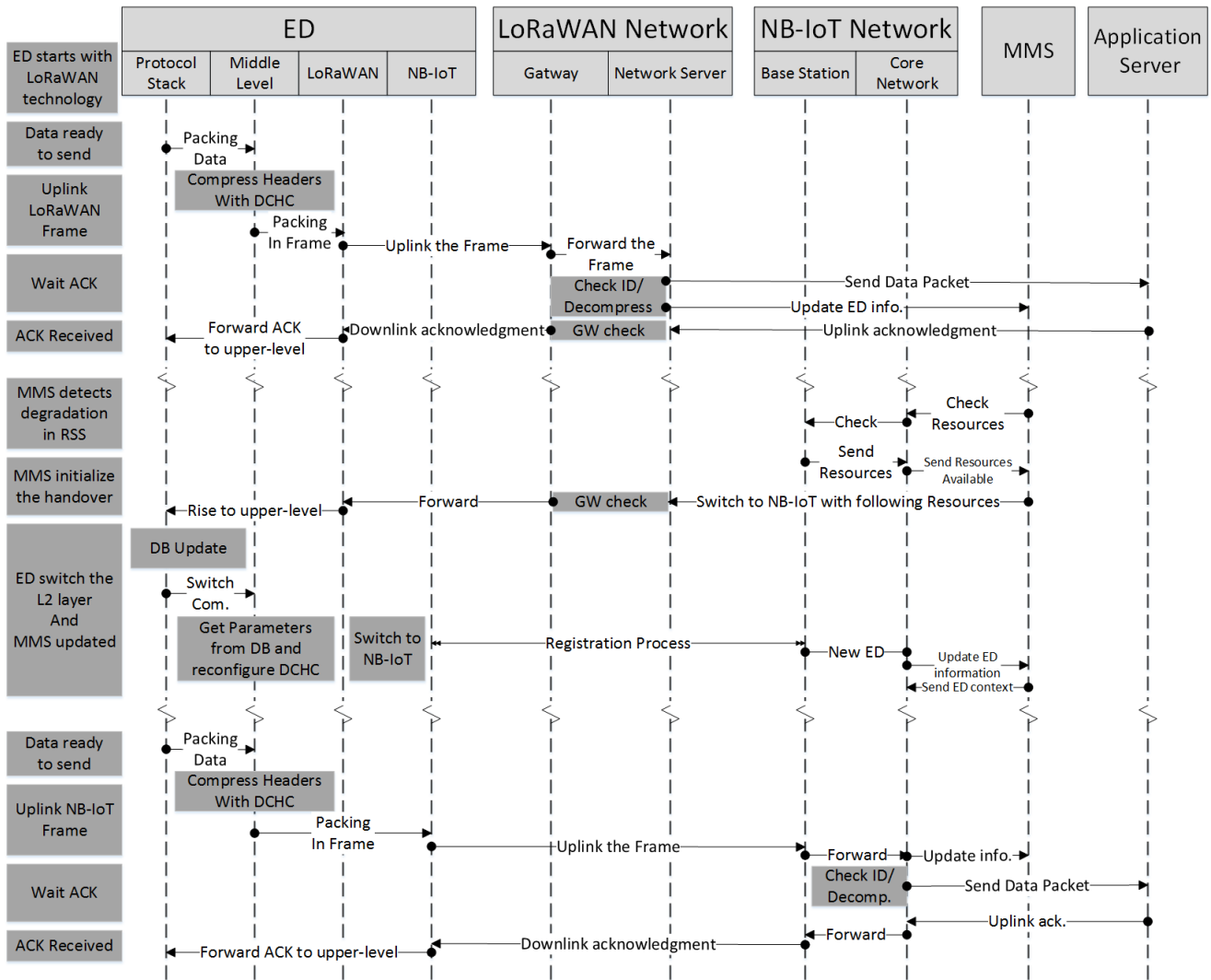


Figure 14: Network initialize Handover from LoRaWAN to NB-IoT

NB-IoT technology. During this step, the TSA pauses the transmission of the running applications and waits to receive an event indicating that the radio is ready. After reconfiguration of DCHC, the protocol stack receives the event, then the registration process begins with the NB-IoT network.

In LPWAN, the ED does not have a disconnection procedure with the previous network. Also, it does not have to inform the network about the handover. Once the device is registered with the NB-IoT base station, the network informs MMS that the switching process was successful. After that, the MMS sends the context of the device used to decompress the received packets and stores them in the core network. Beyond that, when the data ready to send at ED side, the IP packet is compressed and then framed to transmit technology i.e, NB-IoT. Before transmitting, the ED has to resume the connection with the base station, as shown in section IV of [4], then the compressed frame is transmitted. Upon receiving the compressed frame by the core network, it decompresses and forwards the IP packet to the AS.

4.2. ED initialization

Consider that the ED is initially connected to the NB-IoT network. As shown in fig. 15, the device detects the degradation in RSS using the ALES. Then, TSA consults the MMS about available technologies close to it [7]. When MMS receives the request from ED, it queries the LoRaWAN network to verify the available resources. After that, the MMS sends the device information including the frequency, spreading factor, etc, that should be used. Then, the TSA updates the database on the ED, and sends a command to the ALC layer to update DCHC and switch to LoRaWAN technology. ALC verifies the database to obtain the new parameters to reconfigure the DCHC layer. At the end, an event rises to the upper layers indicating that the link layer is ready to use.

After successful registration with LoRaWAN network, the network server updates the MMS with the location and parameters of the ED. In turn, the MMS returns with the context of the ED. At the end of the process, the ED transmits data and waits for acknowledgment.

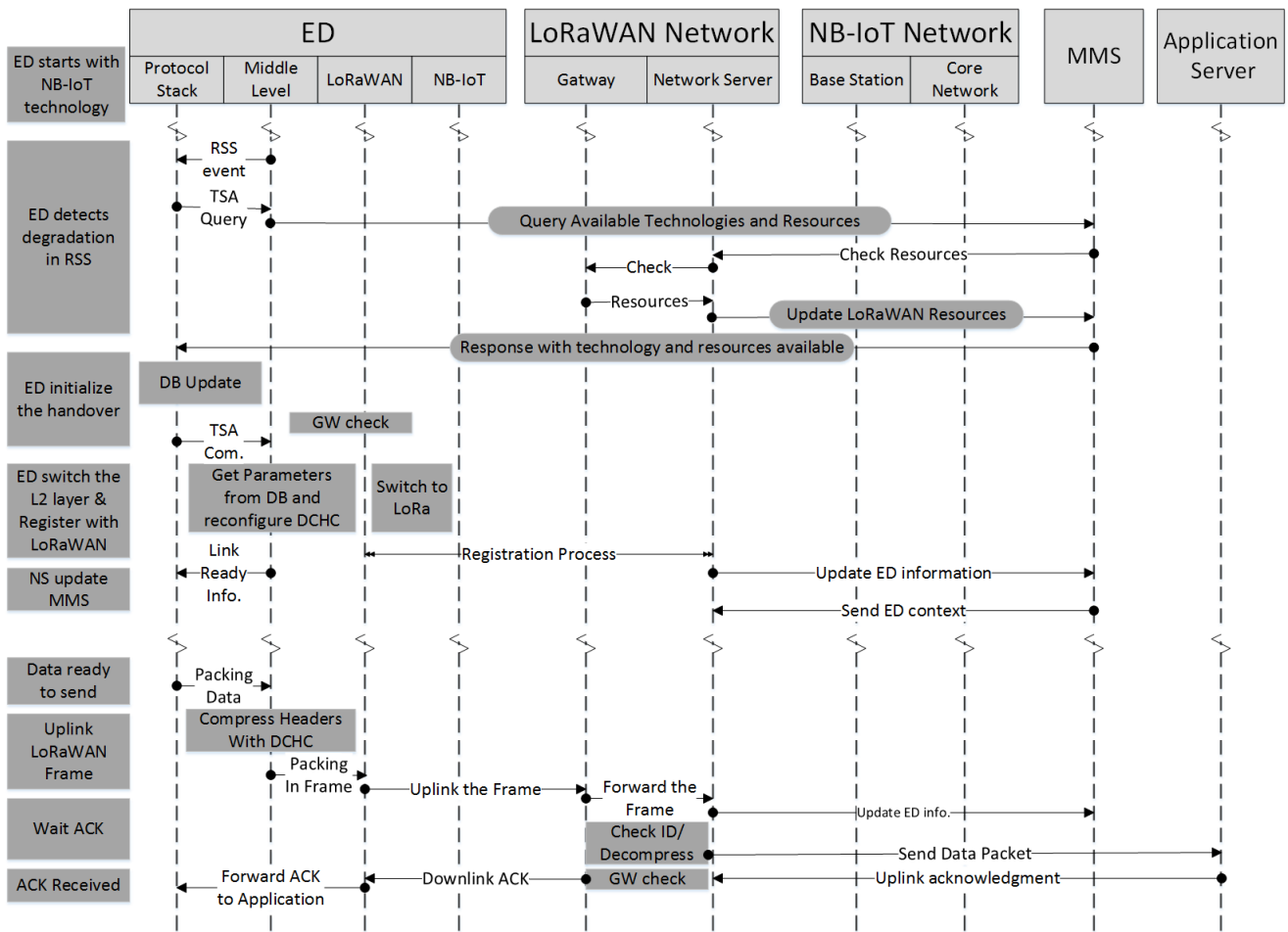


Figure 15: ED initialize Handover from NB-IoT to LoRaWAN

4.3. Connection lost

In some situations the connection may be lost. In such a situation, the LoRaWAN ED repeats the uplink several times, as shown in fig. 7, until it receives an ACK. Meanwhile, the NB-IoT ED returns to the inactive state and listens to incoming beacons from a nearby base station. In either case, the ED consumes a certain amount of time trying to achieve a connection with the currently network used. After the termination of the number of attempts, the EDs that support our proposal seek other networks in the near environment. Therefore, ED begins by checking the networks available in the nearby by activating one of the radios it supports at the lower level. For example, to verify NB-IoT coverage, the device will reconfigure DCHC with the default parameters of the NB-IoT network and activate the radio to listen to incoming beacons. When a network is discovered, the ED begins the registration process. When the registration is successful, the device repeats the failed transmission.

5. Mobility Management between heterogeneous networks

To support continuity of communication after network changes, we propose to use compressed IP-based solution.

But, the context used to C/D IP packet headers must be exchanged and updated as the ED network changes. To test the validity of the proposed framework, we consider the scenario shown in fig. 16, where an ED moves from the HN (NB-IoT) to the VN (LoRaWAN). Independent of the used technology on VN whether it is the same as the home or not, the context of the ED is unknown to this network. Without these contexts, communication cannot be achieved. Therefore, it is required to share and manage the context between networks. In this scenario, we show the MMS functionalities in context management and communication improvement.

5.0.1. Simulation

Based on the work done in NS3 [23] where the NB-IoT network is implemented, we added LoRaWAN [20] to the network. Therefore, we have both LoRaWAN and NB-IoT networks in NS3. In our implementation, we consider that both networks belong to the same network operator. Therefore, we implemented MMS with the network operator as shown in fig. 11. On top of that, we created the ED and we installed on it the MAC layers of the two networks: LoRaWAN and NB-IoT. Then, we added the protocol stack proposed in fig. 8 to the ED [24]. According to the scenario shown in fig. 16, we simulated an ED that is initially con-

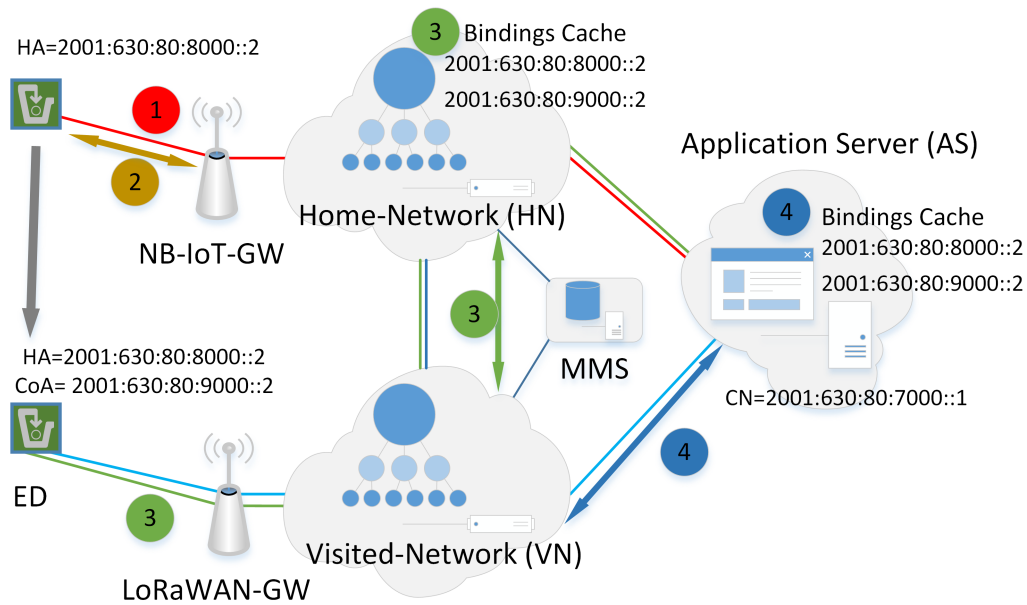


Figure 16: Switch from NB-IoT to LoRaWAN

connected to the NB-IoT network and then switches to the LoRaWAN network as shown in fig. 15.

5.0.2. Communication scenario

Before the ED leaves the coverage of the NB-IoT network, the status of the ED changes from "IDLE" to "connection resume", as shown in fig. 7 and explained in section 4. This is to get ready to transmit the data. The ED packs the data in the form of an IPv6 packet. This packet goes through DCHC [20] that compresses the headers as shown in Table 2. In this table, we illustrate the compression of IP packet headers at the network layer with the context (NLC). As the ED is assigned to the NB-IoT network, the context of the headers is not changed and it is saved in the network. As shown in Table 2, the ED only sends the rule number without any headings. Thus, Rule 1 matches all the used headers [5]. The size of Rule 1 is one byte as explained in section 3 [21]. However, the compressed header is empty, since the matching operator (MO) is ignored.

Then, we consider that during the ED movement, it detects that the RSS value with the NB-IoT GW has decreased. Therefore, ED transmits a message in step two to MMS asking about the available networks in the nearby region that support better communication. After that, MMS contacts LoRaWAN network servers that cover EDs geographical location with its GWs. Then, MMS responds by providing information about the available networks. The format of the reply message is the same as in the table 2 with the source and destination address headers exchanged. During this procedure, MMS updates the context and pre-registrations (at the data link layer) for ED in the LoRaWAN network and sends the new identity to ED in addition to CoA compatible with the network.

Upon receiving the response, and before leaving the NB-

IoT network, the TSA block of the ED sends a command to the middle level layer to change the link layer to LoRaWAN as explained in section 4. At the data link layer, ED takes the new identity and saves the keys for communication with LoRaWAN network. At the network layer level, ED selects one of the CoAs found in the response message and uses it as a new L3 identity (at the network layer) for the link update procedure as shown in step three of fig. 16 (see green line). Then, the ED transmits the first LoRaWAN frame with compressed headers. As shown in Table 3 the CoA is the source address, the AS address is the destination address and the HoA is set in the extension headers. Thus, we divide the table into two parts; the main context in the NLC and the extension context in the extension layer context (ELC). This packet will be named as handover packet, since it is the first packet transmitted after handover between NB-IoT and LoRaWAN.

The shape of the transmitted LoRaWAN frame is shown in fig. 17, where the PHY payload contains the RuleID (1 byte), compressed headers (n bytes), and the data set in the PHY payload. The values sent in Table 3 are set in the Compressed Header field. After decompressing the payload of the LoRaWAN frame, the IP packet is reformed and sent over the network. Following the decompression, the visited operator can use the routing optimization method [5] to communicate directly with AS without going through the NB-IoT network as shown in fig. 16 (see blue color).

5.0.3. Results

In the fig. 18, we illustrate the steps that result from this NS3 simulation. As we can see, in the network that does not support MMS, the ED loses the connection before switching to another network. After several failures while trying to connect to another NB-IoT GW, ED starts checking other

Table 2
IPv6 compression using DCHC at ED side

| Rule 1 | | | |
|---|----|--------------|-------------|
| NLC layer | | | |
| Field | FL | TV | C/D Action |
| Version | 4 | 6 | not-sent |
| DiffServ | 8 | 0x00 | not-sent |
| FL | 20 | 0x000000 | comp-length |
| Length | 16 | None | not-sent |
| NH | 8 | 17 | not-sent |
| HL | 8 | 30 | not-sent |
| SA prefix | 64 | 2001:63:80:8 | not-sent |
| SA addr. | 64 | ::2 | not-sent |
| DA prefix | 64 | 2001:63:80:7 | not-sent |
| DA addr. | 64 | ::1 | not-sent |
| Source Address= 2001:63:80:8::2 (Home Address of ED) | | | |
| Destination Address = 2001:63:80:7::1 (Address of AS) | | | |
| MO = Ignore / DL = Bi / FP = 1 | | | |

Table 3
Handover packet header compressed using DCHC

| Rule 2 | | | |
|---|----|--------------|-------------|
| NLC layer | | | |
| Field | FL | TV | C/D Action |
| Version | 4 | 6 | not-sent |
| DiffServ | 8 | 0x00 | not-sent |
| FL | 20 | 0x000000 | Value-sent |
| Length | 16 | None | Comp-length |
| NH | 8 | 17 | value-sent |
| HL | 1 | 30 | value-sent |
| SA prefix | 64 | 2001:63:80:9 | value-sent |
| SA addr. | 64 | ::2 | value-sent |
| DA prefix | 64 | 2001:63:80:7 | value-sent |
| DA addr. | 64 | ::1 | value-sent |
| ELC layer | | | |
| ED P HoA | 64 | 2001:63:80:8 | value-sent |
| ED HoA | 64 | ::2 | value-sent |
| Care of Address= 2001:63:80:9::2 (New HA of ED) | | | |
| Home Address= 2001:63:80:8::2 (Previous HA of ED) | | | |
| Destination Address = 2001:63:80:7::1 (Address of AS) | | | |
| MO = Ignore / DL = Bi / FP = 1 | | | |

supported technologies. In our simulation, we configured the device to try to listen to the incoming system information five times from a nearby NB-IoT GW. Therefore, the ED took 16 seconds in our simulation in the state of connection lost, as shown in fig. 18, before attempting to change layer L2 to LoRaWAN. While, in MMS network support, the server sends to ED the information required for the ED to switch to the LoRaWAN network. Then, the ED returns to the "IDLE" state and changes the L2 layer and is configured with the LoRaWAN address. Then, the ED transmits the data. This process costs 158 ms in our simulator to change, obtain database values and configure. But, we will not consider the moment of change, since it runs on a PC and not on a real ED.

In this experiment, MMS contributes to the handover in

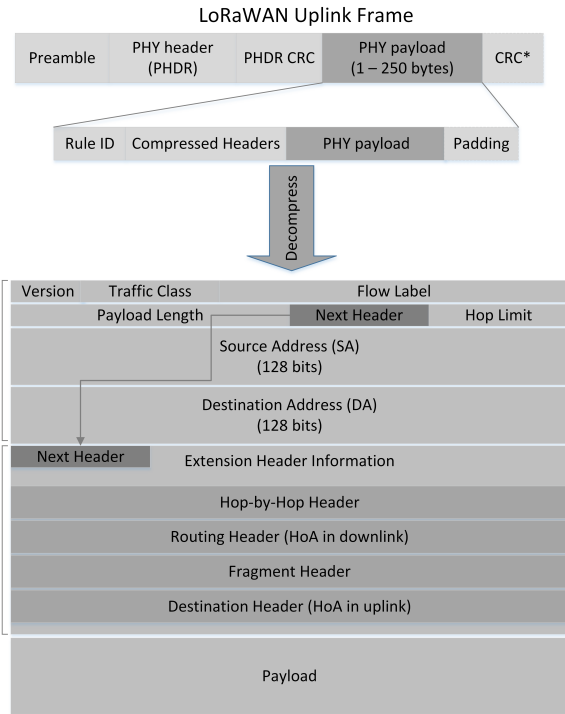


Figure 17: Decompression of LoRaWAN Frame

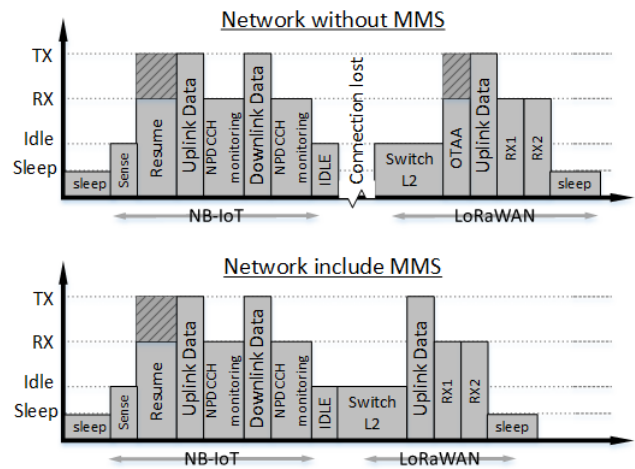


Figure 18: Switch from NB-IoT to LoRaWAN

the data link layer, while compressed LMIPv6 contributes to the continuity of the communication with AS after the ED identity changes at L3 layer. In summary, the proposed framework saves time of the ED consumed in the lost connection state and prevents loss of connection. Actually, in a real case, the speed of the ED installed in the truck should be considered. Therefore, in a network without MMS, a delay of 16 seconds followed by 6 seconds to register with the LoRaWAN network or 10 seconds with the NB-IoT network is unacceptable with high speeds. By delimiting C/D management with MMS, the context will no longer be dispersed among the different servers of the operators. Centralizing

these contexts on a server facilitates its administration and updating. In addition, it frees the ED from the responsibility of sharing the context with the operators. This preserves its message rate, especially when IoT constrained technologies such as LoRaWAN and SigFox are used. In addition, the use of the IPv6 communication protocol allows the handover while the home-IP address is reserved within the package. This contributes to the optimization of routing and saves the bandwidth required to route packets with the home operator. In summary MMS contributes to:

- Improving the DCHC mechanism and managing context sharing.
- Improving the handover in layer L3.
- Saving bandwidth between networks and ED messages rate.

6. Mobility Management within the technology

6.1. Scenario in case of LoRaWAN network

As mentioned in the section 2, the LoRaWANv1.1 specification supports two types of activations to register the ED on the network: ABP and OTAA. In addition, two types of collaborations, to support the mobility of EDs, are defined in LoRaWANv1.1: passive and handover [7]. We said that OTAA was preferable for mobility, since after the loss of connection, ED can register with other networks in [5]. While the ED was activated through the ABP process, its mobility was limited within the registered network and passive roaming. In OTAA, the ED can be assigned a new identity at L2 layer (that is, DevAddr) when moving to networks other than Home. While, the ABP ED identity remains fixed. In communication, the OTAA process takes up to 6 seconds for ED to be assigned to a network. While the ED activated through the ABP process can directly transmit the data to the network without a registration delay. Collaborations were defined to support ED roaming between different operators or network servers. In passive roaming, the DevAddr will not change. While, with the handover collaboration, the identity of the ED have to be changed. Therefore, ABP ED can only be moved to a visited network server with passive roaming collaboration with its home network server. While EDs that support the OTAA process can be moved to visited network servers with either types of collaboration with its home network server.

6.1.1. Objective of the experiment

Due to that ABP process is personally performed by a developer, ED cannot change its DevAddr on air by exchanging keys as in OTAA. This prevents this ED from roaming to the Visited network server with the handover collaboration. But we will try to study whether we can assign a new DevAddr for this ED using MMS and MAC commands. Therefore, the ED that supports the ABP activation process will be assigned a new DevAddr and network session key, thanks to MMS, without exchanging keys with the Visited network server on

air. But, the key and identity will be sent to the ED in the last downlink message before the handover. On success, the ED that was activated using ABP can be able to move to any network server independently of the collaboration between them.

6.1.2. Testbed of the experiment

To test the proposed framework, a testbed was built to measure the improvement in terms of time. Our testbed consists of three network operators, as shown in fig. 19. Using Raspberry Pis and LoRaWAN shields, with the help of the LoRaServer [25] project, we created the LoRaWAN network and implemented the proposed stack in the ED. The network servers are involved in the experiment are: H-NS, V-NS (X) and V-NS (Y). GWs A and B belong to H-NS, C and D belong to V-NS (X) and E and F belong to V-NS (Y). The distance between every two GWs that belong to the same operator is 500 m, that is, A and B, while it is 800 m between GWs of different operators, that is, B and C. We consider that the transfer between H-NS and V-NS (X) is a passive collaboration, while it is a handover collaboration with V-NS (Y). The moving ED is based on an Raspberry Pi 2 equipped with a LoRaWAN shield [26]. It was programmed as OTAA during the first test and ABP during the second one. In addition, the Wireshark program is used to measure the time between each two components of the network. To bring the implementation closer to reality, we used the Pfsense [27] project to manage bandwidth, add delays and manage routes between different components. Otherwise, the time between the components is less than 1 ms. The Join Server and MMS run on the PC that is connected to the same network. The ED is associated with the H-NS at startup. Next, we will measure the time required for a complete communication procedure that begins when an ED sends an uplink LoRaWAN frame to AS followed by an acknowledgment sent from AS to ED. In each position, we compare the time required for a complete procedure using the LoRaWANv1.1 network without MMS and the same procedure when MMS is installed on the network. During the measurements, we configure the device to use a payload size of 155 bytes, the Broadcast Factor (SF) is set to 10 and the bandwidth is set to 125 kHz. The 155-byte payload consists of 40 bytes representing uncompressed LMIPv6, 8 bytes for UDP, 4 bytes, 1 byte for RuleID, 1 to 5 bytes for compression headers, and the remaining bytes are data.

6.1.3. Results

In position P1, as shown in fig. 19, the ED is under GW(A) coverage. As shown in Table 4, it transmits a LoRaWAN frame to GW(A). Then, the LoRaWAN frame is forwarded to the H-NS. In the H-NS, the identity is verified, then the LoRaWAN frame data is unpacked and sent to AS. In this position, ED communicates with the H-NS, so MMS has no contributions with the registration (ABP or OTAA). Therefore, the ED can deliver data to AS. The total communication time was 2162 ms.

In position P2, as shown in fig. 19, the ED is under

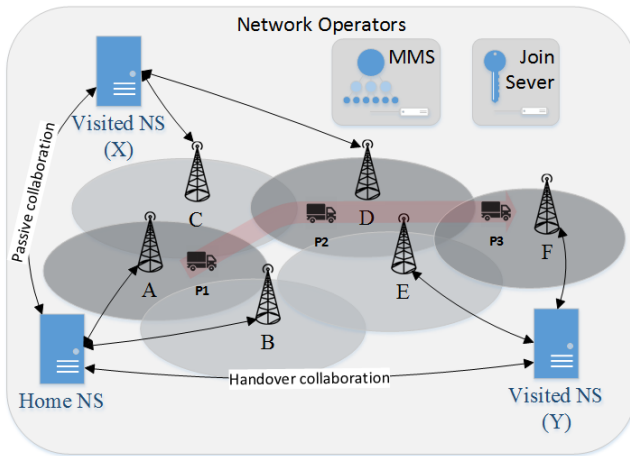


Figure 19: Proposed LoRaWAN network

the coverage of GW(D). As shown in Table 4, it transmits a LoRaWAN frame to GW(D). This transmission is 1,50938 seconds. Then, GW (D) forwards the LoRaWAN frame received to V-NS(X). The link between GW(D) and V-NS(X) lasts 50 ms to resend the received LoRaWAN frame. Since the collaboration with the H-NS is passive, the V-NS(X) verifies only the identity of the device in the L2 layer and sends the LoRaWAN frame back to the H-NS. Because it is not necessary to change the identity, the ED (ABP or OTAA) can deliver data to AS. The total process is achieved in 2738 ms. This additional time due to the cost of the link between the two servers.

In position P3, as shown in fig. 19, the ED is under GW (F) coverage. As shown in Table 4, when the network does not support MMS, the ED that was activated using ABP can no longer deliver data to AS. In addition, when ED uses OTAA, the LoRaWAN frames are dropped by V-NS(Y) and unacknowledged. Then ED detects that there is no coverage while acknowledgement not received. Therefore, this ED performs the registration process and retransmits LoRaWAN frame. The total process costs 11573 ms to deliver the data to AS. While, in an MMS-compatible LoRaWAN network, the device obtains a new identity and keys before losing the connection with the V-NS (X). As shown in the Table 4, the ED (ABP or OTAA) could deliver data to AS after 2625 ms.

Therefore, MMS has contributed by allowing EDs with the ABP process to move through operators independently of the collaboration, passive or handover. In addition, it saves the time of OTAA ED to deliver the data to the AS.

6.2. Scenario in case of NB-IoT network

In NB-IoT and as we showed in [4], the ED is registered with the GW. After transmission, ED returns to the IDLE state. Meanwhile, the NB-IoT GW saves the parameters of the connection with the ED. These parameters allow the ED to resume the connection with less time than the registration process. But, when ED moves from one GW to another, these parameters are lost and not exchanged between the GWs. However, in the NB-IoT standard, the "X1" link

Table 4
Measurements of the latency

| | Without MMS | | With MMS |
|---------------------------------------|-------------|----------|----------|
| | ABP | OTAA | ABP |
| At position P1 | | | |
| Transmit Data (Truck to A) | 1509 ms | | |
| Process and forward (A to H-NS to AS) | 480 ms | | |
| ACK (AS to Truck) | 173 ms | | |
| Total | 2162 ms | | |
| At position P2 | | | |
| Transmit Data (Truck to D) | 1510 ms | | |
| Process and forward (D till AS) | 608 ms | | |
| ACK (AS to Truck) | 620 ms | | |
| Total | 2738 | | |
| At position P3 | | | |
| Transmit Data (Truck to F) | 1510 ms | | |
| Process and forward (F till AS) | - | 475 ms | |
| Waiting ACK | 2000 ms | | - |
| Register with Y | - | 5438 ms | - |
| Repeat Transmission | 1510 ms | | - |
| ACK (AS to Truck) | - | 640 ms | |
| Total | - | 11573 ms | 2625 ms |

between the GWs still exists. But in the standard, they eliminate the transfer of ED between two GWS corresponding to the same network to avoid complexity. Therefore, this motivated us to study possible improvements in the communication with MMS in order to decrease the duration of the registration process.

Until now, it is not possible to detect the exact GW to which the ED will be connected unless we use GPS. As the GW covers a large area, we can use the geographical location of the ED to know about the GW covering that area. Also, it would be easier to use a GPS for the same purpose. In both cases, MMS will connect to the NB-IoT network and obtain information about the GW that covers the expected area through which the ED will pass. Then, MMS will send this information to ED before losing the connection with current GW. Depending on the ED, it may decide to change the connection or not. In the MMS message, ED receives information that contains synchronization, MIB and SIB. Then, ED can deal with this information and sends a comment for ALC to update the L2 layer. Then, the ED starts directly with the RAP.

6.2.1. Simulation

Since an NB-IoT network is not available to perform our tests, we used the NS3 software to implement our test scenario. NS-3 has a complete LTE module that includes EPC control and data plane, an end-to-end data control plane protocol stack, a physical layer data error model, radio propaga-

- [4] W. Ayoub, A. E. Samhat, F. Nouvel, M. Mroue, and J. Prévotet, "Internet of mobile things: Overview of lorawan, dash7, and nb-iot in lpwans standards and supported mobility," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2018.
- [5] W. Ayoub, F. Nouvel, A. Samhat, M. Mroue, and J. Prévotet, "Mobility management with session continuity during handover in lpwan," *IEEE Internet of Things Journal*. doi: 10.1109/JIOT.2020.2985925.
- [6] B. S. Ghahfarokhi and N. Movahhedinia, "A survey on applications of iee 802.21 media independent handover framework in next generation wireless networks," *Computer Communications*, vol. 36, no. 10, pp. 1101 – 1119, 2013.
- [7] W. Ayoub, M. Mroue, A. E. Samhat, F. Nouvel, and J.-C. Prévotet, "Schc-based solution for roaming in lorawan," in *Advances on Broad-Band Wireless Computing, Communication and Applications* (L. Barolli, P. Hellinckx, and T. Enokido, eds.), (Cham), pp. 162–172, Springer International Publishing, 2020.
- [8] W. Ayoub, M. Mroue, F. Nouvel, A. E. Samhat, and J. c. Prévotet, "Towards ip over lpwans technologies: Lorawan, dash7, nb-iot," in *2018 Sixth International Conference on Digital Information, Networking, and Wireless Communications (DINWC)*, pp. 43–47, April 2018.
- [9] W. Ayoub, F. Nouvel, S. Hmede, A. E. Samhat, M. Mroue, and J. Prévotet, "Implementation of schc in ns-3 and comparison with 6lowpan," in *2019 26th International Conference on Telecommunications (ICT)*, pp. 432–436, 2019.
- [10] LoRa Alliance, <https://www.lora-alliance.org/lorawan-for-developers>, *LoRaWAN v1.1 Specification*, october 11, 2017 ed.
- [11] 3rd Generation Partnership Project (3gpp), http://www.3gpp.org/ftp/Specs/archive/45_series/45.820/, *Cellular system support for ultra-low complexity and low throughput Internet of Things (CIoT) (Release 13)*, 2015 ed.
- [12] L. Feltrin, G. Tsoukaneri, M. Condoluci, C. Buratti, T. Mahmoodi, M. Dohler, and R. Verdone, "Narrowband iot: A survey on downlink and uplink perspectives," *IEEE Wireless Communications*, vol. 26, pp. 78–86, February 2019.
- [13] A. Adhikary, X. Lin, and Y. . E. Wang, "Performance evaluation of nb-iot coverage," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, pp. 1–5, Sep. 2016.
- [14] R. Ratasuk, B. Vejlgaard, N. Mangalvedhe, and A. Ghosh, "Nb-iot system for m2m communication," in *2016 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pp. 428–432, April 2016.
- [15] M. Lauridsen, I. Z. Kovacs, P. Mogensen, M. Sorensen, and S. Holst, "Coverage and capacity analysis of lte-m and nb-iot in a rural area," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, pp. 1–5, Sep. 2016.
- [16] B. Vejlgaard, M. Lauridsen, H. Nguyen, I. Z. Kovacs, P. Mogensen, and M. Sorensen, "Coverage and capacity analysis of sigfox, lora, gprs, and nb-iot," in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, pp. 1–5, June 2017.
- [17] A. Minaburo, L. Toutain, C. Gomez, D. Barthel, and J.-C. Zúñiga, "Static Context Header Compression (SCHC) and fragmentation for LPWAN, application to UDP/IPV6," Internet-Draft draft-ietf-lpwan-ipv6-static-context-hc-21, Internet Engineering Task Force, July 2019. Work in Progress.
- [18] A. Minaburo, L. Toutain, and R. Andreasen, "LPWAN Static Context Header Compression (SCHC) for CoAP," Internet-Draft draft-ietf-lpwan-coap-static-context-hc-11, Internet Engineering Task Force, Oct. 2019. Work in Progress.
- [19] W. Ayoub, A. E. Samhat, M. Mroue, H. Joumaa, F. Nouvel, and J.-C. Prévotet, "Technology selection for IoT-Based smart transportation systems," in *International Workshop on Vehicular Adhoc Networks for Smart Cities (IWVSC'2019) (IWVSC'2019)*, (Paris, France), Nov. 2019.
- [20] W. Ayoub, F. Nouvel, S. Hmede, A. E. Samhat, M. Mroue, and J.-C. Prévotet, "Implementation of SCHC in NS-3 simulator and comparison with 6LoWPAN," in *2019 26th International Conference on Telecommunications (ICT) (ICT 2019)*, (Hanoi, Vietnam), Apr. 2019.
- [21] A. Minaburo, L. Toutain, and C. Gomez, *LPWAN Static Context Header Compression (SCHC) and fragmentation for IPv6 and UDP*. IETF lpwan Working Group, Dec. 22, 2017. <https://tools.ietf.org/html/draft-ietf-lpwan-ipv6-static-context-hc-09>.
- [22] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "Overview of cellular lpwan technologies for iot deployment: Sigfox, lorawan, and nb-iot," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 197–202, March 2018.
- [23] K. Lin and S. Yang, *Simulating NB-IoT Random Access: Using Ns-3 as an Example*. National Tsing Hua University, 2018. <https://books.google.ro/books?id=R1k8zAEACAAJ>.
- [24] W. Ayoub, "Implementation of dschc framework over nb-iot and lorawan networks on ns3." https://gitlab.insa-rennes.fr/Wael.Ayoub/Network_architecture_support_IPv6_on_NB-IoT_and_LoRaWAN/tree/master, June 2019.
- [25] cablelabs, sidnfonds, and acklio, "Loraserver project." <https://www.loraserver.io/>, Accessed on 13 Nov 2018.
- [26] Dragino, *Lora/GPS HAT for Raspberry Pi*. Media Wiki for Dragino Project, 2018. https://wiki.dragino.com/index.php?title=Lora/GPS_HAT.
- [27] R. Botelho, L. O. O. Souza, J. Pingle, J. Dillard, S. Beaver, and M. Smith, "Open source security firewall from netgate." <https://www.pfsense.org/>, Accessed on 5 Feb 2019.
- [28] M. El Soussi, P. Zand, F. Pasveer, and G. Dolmans, "Evaluating the performance of emtc and nb-iot for smart city applications," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–7, May 2018.
- [29] K. Lin and S. Yang, *Simulating NB-IoT Random Access*. National Tsing Hua University, Aug 2019. <https://github.com/a3794110/ns-3-NB-IoT>.