Understanding and Characterizing Transmission Times for Compressed IP packets over LoRaWAN

Rodrigo Muñoz Lara
Dept. of Electrical Engineering
Universidad de Chile
Santiago, Chile
rmunozlara@ing.uchile.cl

Sandra Céspedes
Dept. of Electrical Engineering
Universidad de Chile
Santiago, Chile
scespedes@ing.uchile.cl

Abdelhakim Hafid
Dept. of Computer Science and Op. Research
University of Montreal
Montreal, QC, Canada
ahafid@iro.umontreal.ca

***Abstract*—The Low Power Wide Area (LPWA) networks are expected to enable the massive connectivity of small and constrained devices to the Internet of Things. Due to the restricted nature of both end devices and network links, LPWA technologies employ network stacks that often do not define an interoperable network layer; instead, application data is usually placed directly into technology-specific layer-two frames. Besides not being able to run standard IP-based protocols at the end device, the lack of an IP layer also causes LPWA segments to operate in an isolated manner requiring middle boxes to interface non-IP LPWA technologies with the IP world. IETF is working to standardize a compression scheme, called *Static Context Header Compression* (SCHC), which will allow compressing the headers of IPv6 and UDP for LPWA networks, in a way that the end device is enabled with IP-based communications. In this paper, we focus on the LoRa/LoRaWAN technology and describe the way in which the selection of the compression rules in SCHC will translate into specific transmission delays. We conduct a study of the expected transmission times, namely the Time on Air (ToA), and propose two additional metrics to understand and characterize the impact of compression over the resultant delay of IP-based communications over LoRaWAN. Our work concludes that there is a non-linear relationship between Time on Air and the percentage of compression of each rule.**

***Index Terms*—Compression, IPv6, LoRaWAN, LPWAN, SCHC**

I. Introduction

With the introduction of the *Long Range Wide Area Network* or *LoRaWAN* technology, as a representative technology of the LPWA category, many solutions for the Internet of Things have been able to cover greater distances and to consume less energy at the end devices. The wider coverage and lower consumption come at the cost of very low bandwidth, variable and restricted MTUs, and a low duty cycle —which in most cases is less than 1% [1]— resulting in a limited number of messages per hour.

As illustrated in Fig. 1a, the LoRaWAN architecture considers an element, called the Network Server, responsible for being the bridge between end-devices and the Internet. Due to the restricted nature of both end devices and network links, LPWA technologies, and in particular LoRaWAN, employ network stacks that often do not define an interoperable network layer; instead, application data is usually placed directly into layer-two frames. Besides not being able to run standard IP-based protocols at the end device, the lack of an IP layer also causes LPWA segments to operate in an isolated manner requiring middle boxes to interface non-IP LPWA technologies with the IP world.

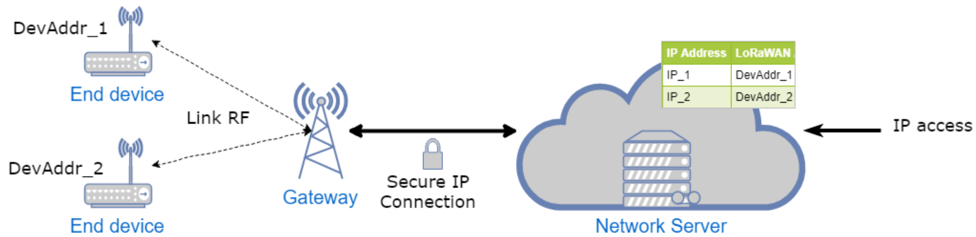
In the past, it has been recognized that the advantage of directly connecting LPWA end devices to the Internet, via a common IP layer, lies in the nodes being able to use mobility, segmentation, and transport features that IPv6 delivers today as a network layer protocol [2] [3] [4]. The use of IP also enables the use of common components for management and security, as well as shared application profiles. One additional benefit is the possibility to ease the transition of an IoT solution from one LPWA technology to another. IPv6, in addition to providing a network protocol to interconnect different devices, also enables higher-layer IP-based protocols to interoperate with other IP-based systems. For example, the use of Web-based communications is a possible way of delivering interoperability. The term *Web of Things* [5] can be used to unite all efforts that aim at creating interoperable systems that involve a large number of integrated devices while relying on web communications, as it was achieved with the successful World Wide Web.

With the goal of interoperability in mind, the *Internet Engineering Task Force (IETF)* LPWAN working group has worked on a standard, called *Static Context Header Compression* (SCHC), which allows compressing IPv6 and UDP headers (and soon the Constrained Application Protocol or CoAP) for LPWA networks, enabling the use of IP-based communications directly at the end device [6]. The SCHC proposal operates on the base of static contexts; this means that the values in the headers of IPv6 and UDP packets, to be compressed, are known or inferable from information of other headers. Figure 1b shows the different elements of the LoRaWAN architecture when they are enabled to support IPv6 through the use of the proposed SCHC standard.

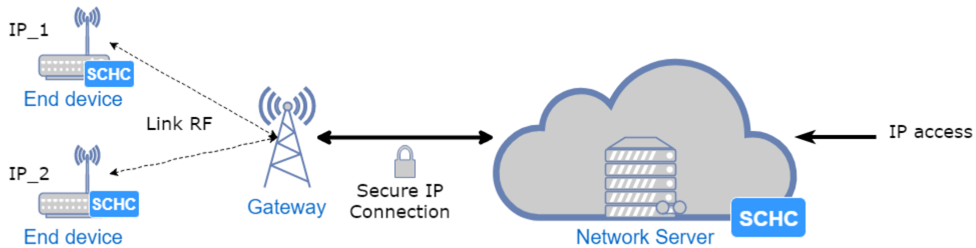
Within a SCHC context, one can find different rules that indicate the behavior of the compressor/decompressor for each IPv6 and UDP header. The selection of these rules directly

This work was supported in part by CONICYT Chile, Project RETRACT ERANET-LAC under Grant ELAC2015/T10-0761

978-1-7281-3955-5/19/$31.00 ©2019 IEEE



(a) No IP support at the end-device



(b) IPv6 support in end-device with SCHC

Fig. 1. LoRaWAN network architecture

impacts the number of bytes that make up the compressed packet, and therefore, the time it takes to transmit each compressed packet on the wireless medium. The transmisson time has been typically called the Time on Air (ToA) [1], [7]. This paper proposes a methodology to understand the behavior of ToA and characterize its performance under different operating conditions, i.e., variable Spreading Factors, of the LoRaWAN link.

A. Motivation

This work seeks to establish the behavior of ToA when increasing or decreasing the SCHC packet compression ratio: does it vary linearly with the percentage of compression given LoRa Spreading Factors? We expect to help network designers or operators to make an informed selection of a compression rule given its ToA behavior is known. For example, being able to select a set of rules knowing ToA remains constant no matter the rule used within the set. In the case IoT applications contain timers-based state machines, an increase in ToA may activate states designed to deal with packet losses; however, this increase may instead be the effect of a longer delay due to a change of the compression rule. In the event that the chosen rule is fixed, it is still necessary to identify how the changing nature of LoRa Spreading Factors entails a variation in ToA of the packets compressed with that specific rule.

Additionally, we introduce two new metrics to characterize the Time on Air. One metric allows to determine the compression rules for which the modification in the Spreading Factors results in a minimum change of ToA. This would be beneficial for applications that are sensitive to delay variations, allowing to identify the modifications in the compression ratio that do not involve major changes in the transmission time. The other metric is to determine the maximum variation of ToA when the compression percentage is modified for a given Spreading Factor. This is useful for applications that have multiple data streams, since each stream will be associated with a different rule and therefore a different compression ratio. Thus, although the Spreading Factor is stable, these flows may experiment different delays.

B. Previous works

The transmission of IP packets over LPWA networks has received the interest of academia and industry in recent years. In [2], LoRaWAN, DASH7, and NB-IoT technologies are compared based on their ability to support the IP protocol through different existing standard solutions, such as 6LowPAN and 6Lo, 6TiSCH, RObust Header Compression (ROHC), LoWPAN IPHC/NHC, Routing Protocol for Low-power and lossy networks (RPL), CoAP, and SCHC. The work concludes that CoAP and SCHC provide the best approach that can be taken as a base solution for LPWA technologies. The authors in [8] present a survey of the existing standards and ongoing efforts to provide an interoperable stack for the Internet of Things.

The reduced bandwidth and small payload of LPWA networks make challenging the adoption of dynamic contexts for compression; nevertheless, in [9], the authors propose the use of dynamic contexts with a fictitious mapping to flows that are not known beforehand. This technique works with headers that are copied from the original request and inserted into the corresponding response. When an IP packet arrives at the gateway, it applies a RuleID in which the unknown header H takes a temporary value previously agreed between the gateway and the final node. When rebuilding the IP packet, the final node delivers the header H with a dummy value to the application layer. In the response, the application uses the same dummy value delivered in the request. When the gateway receives the SCHC packet, it reconstructs the IP packet assigning the original stored value (i.e., the one obtained from the mapping between the original value and the dummy value) to the header *H*.

On the implementations side, the IETF LPWAN working group has been the host of several hackatons promoting the development of OpenSCHC, an open source implementations of SCHC [10]. OpenSCHC is based on Python and MicroPython, and follows udpates adopted by the LPWAN working group in the SCHC proposal [6]. Nonetheless, the performance evaluation of this implementation has been left open to the community, since the code is still in constant development and to date only provides validation of basic functionality.

None of the aforementioned works has measured or proposed metrics to evaluate the delays generated by SCHC when operating over LoRaWAN with different Spreading Factors, together with considering the high number of "combinations" generated by the rules that will be applicable according to the proposed SCHC standard.

The remainder of this work is organized as follows. In Section II we introduce the mode of operation of SCHC, which is fundamental to understand this work. Section III describes the methodology to conduct this study, both for the use of rules to calculate compression percentages and the description of the new proposed metrics. In Section IV, we present and discuss the results of the performance evaluation. Section V concludes the paper.

II. Static Context Header Compression (SCHC)

The proposed SCHC standard provides compression and fragmentation functionalities for LPWA technologies. The current Internet-draft [6] defines a *context* in which *rules* are stored. Each rule has a series of *descriptors*, named *Field Descriptions*, that define how a specific header should be compressed or decompressed. Figure 2 shows the concepts of Context, Rule, and Descriptor, together with the parameters of each Field Description.

A Field Description is made up of seven fields described below:

- **Field ID (FID)** defines the protocol and header (such as UDP and Destination Port) to be evaluated by the descriptor.
- **Field Length (FL)** represents the bit length of the header.
- **Field Position (FP)** defines which header is evaluated (first, second, third or fourth occurrence) when there are several occurrences on the same packet. Default value is 1 (first occurrence).
- **Direction Indicator (DI)** indicates the direction of the packet. Possible values are *Up*, *Dw*, and *Bi*, to apply the descriptor in uplink, downlink, and both directions, respectively.
- **Target Value (TV)** is used for comparisons with the value of the header that comes in the packet.
- **Matching Operator (MO)** is the operator used to match the header value and the Target Value (TV). MO is only used during the compression phase.

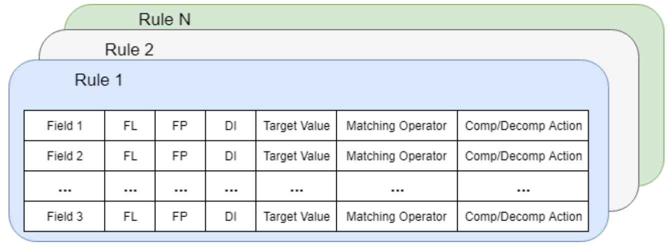


Fig. 2. SCHC Context

- **Compression Decompression Action (CDA)** describes the compression or decompression process that must be performed after all MOs have been evaluated.

When sending a UDP/IPv6 packet, every packet header is compared against each context rule. If the matching operation of all **MO** is **True** when applied to a given context rule, such a compression rule is chosen and the **Compression Decompression Action (CDA)** is applied to each header. If any of the operations of the **MO** results **False**, the rule is discarded and the comparisons continue. The process ends when a rule is chosen and all **CDA** are executed, after which a SCHC packet, like the one illustrated in Fig. 3, is created.

In the SCHC packet, *RuleID* identifies the selected rule and it must be the same ID as the decompression process. The *compression Residue* results from applying all MOs to the packet. *Payload* corresponds to the payload of the upper-layer protocol, e.g., UDP payload.

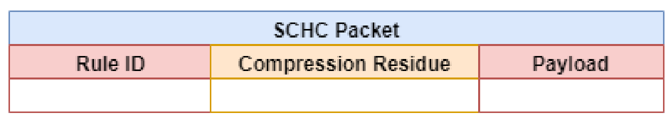


Fig. 3. Structure of a SCHC package

III. Characterizing compression percentages and transmission times

A. Deriving SCHC's compression percentages

Each rule within a SCHC context is based on a finite number of *Field Descriptions*. In total there are 12 descriptors for an UDP/IPv6 packet: 8 descriptors for IPv6 headers (i.e., assuming a basic IPv6 packet) and 4 more descriptors for UDP. Taking the 12 descriptors and combining all the possibilities of compression, one can build a table with 4096 possible rules (2^{12} rules). Fig. 4 illustrates the possible combinations, where 0 indicates that the corresponding header is not compressed (i.e., CDA equal to *value-sent*) and 1 indicates the header should be compressed (i.e., CDA equal to *not-sent*).

Given the following conditions:

- **IPv6.version**: It is always 6 for IPv6, so it can be deducted by the receiver.
- **IPv6.payloadLength, UDP.length** and **UDP.checksum** can be calculated by the decoder.
- **IPv6.nextHeader** is always 0x11 for an UDP/IPv6 packet, so its value can be omitted.

RuleName	RuleID	IPv6.version	IPv6.trafficClass	IPv6.flowLabel	IPv6.payloadLength	IPv6.nextHeader	IPv6.hopLimit	IPv6.sourceAddress	IPv6.destinationAddress	UDP.sourcePort	UDP.destinationPort	UDP.length	UDP.checksum	
Rule 0	0	0	0	0	0	0	0	0	0	0	0	0	0	4096 rules
Rule 1	1	0	0	0	0	0	0	0	0	0	0	0	1	
Rule 2	2	0	0	0	0	0	0	0	0	0	0	1	0	
Rule 3	3	0	0	0	0	0	0	0	0	0	0	1	1	
:	:	:	:	:	:	:	:	:	:	:	:	:	:	
Rule N	N	1	1	1	1	1	1	1	1	1	1	1	1	

Fig. 4. Grouping of rules according to compressed header

- In the uplink direction, the **IPv6.hopLimit** parameter can be considered 0xff since there are no network elements that can modify its value between the SCHC C/D module at the end device and the SCHC C/D module at the Network Side.
- In the uplink direction, if the end device has only one network address assigned, the header **IPv6.sourceAddress** can be considered a static parameter.

Then, the number of headers decreases to 6 resulting in 64 possible rules. The compression percentages for each rule can be obtained considering a total of 48 bytes, from IPv6 and UDP headers, and the evaluation of each rule with the corresponding header sizes. It should be noted that even if a compression of all headers is achieved, a 100% of compression is not feasible since, according to [6], at least the Rule ID should be sent.

The final size of the SCHC packet is therefore the sum of the bits in the Compression Residue, $ComRes$, the bits in the RuleID, $Rule$, and the UDP Payload, $UDP_{payload}$, as indicated in Equation 1.

$$SCHC_{length} = RuleID + ComRes + UDP_{payload}. \quad (1)$$

B. Proposed performance metrics

1) Time on Air (ToA): it is defined as the time it takes for the transmitter to insert a packet into the wireless medium [11]. It is calculated considering the length of the packet in bits and the bit duration according to the Spreading Factor employed at the time of transmission. ToA is calculated as follows [7]:

$$ToA\,(ms) = T_{preamble}\,(ms) + T_{payload}\,(ms)\,, \quad (2)$$

where:

$$T_{preamble}\,(ms) = (n_{preamble} + 4.25) \cdot T_s\,(ms)\,, \quad (3)$$

$$T_{payload}\,(ms) = n_{payload} \cdot T_s\,(ms)\,, \quad (4)$$

$$T_s\,(ms) = \frac{2^{SF}}{BW} \cdot 1000, \quad (5)$$

where, for LoRaWAN [12], $n_{preamble} = 8$, SF ranges from 7 to 12, and the bandwidth, BW, can be 125 KHz, 250 KHz, and 500 KHz.

In the scenario where SCHC compression is used, $n_{payload}$ is calculated as follows:

$$n_{payload} = 8 + max\left[C \cdot (CR + 4)\,; 0\right], \quad (6)$$

$$C = \left\lceil \left(\frac{8PL - 4SF + 28 + 16CRC - 20H}{4\,(SF - 2DE)} \right) \right\rceil, \quad (7)$$

$$PL = MHDR + FHDR + FPort + SCHC_{length} + MIC, \quad (8)$$

where:

- PL: PHY layer payload. According to Equation 8, the PL size corresponds to the sum of the MAC header field size $MHDR$ (1 byte), the Frame header field size $FHDR$ (7 to 22 bytes), the Field port size $FPort$ (1 byte), the MAC payload size corresponding to the SCHC packet size $SCHC_{length}$, and the Message Integrity Code size MIC (4 bytes).
- SF: Spreading Factor with values from 7 to 12
- CRC: 1 when cyclic redundancy check is enabled, 0 otherwise (1 by default in LoRaWAN)
- H: 1 for Implicit Header, 0 for Explicit Header (0 by default in LoRaWAN)
- DE: 1 when Adaptive Data Rate is enabled, 0 otherwise.
- CR: Coding Rate with values $CR = \{1, 2, 3, 4\}$ (1 by default in LoRaWAN)

2) Time on Air variation (ΔToA): it is defined as the subtraction between ToA calculated for the lowest percentage of compression (PC) and ToA calculated for the highest PC given a Spreading Factor (SF). ΔToA is calculated as follows:

$$\Delta ToA_{SF} = ToA_{PC_{min}} - ToA_{PC_{max}} \quad (9)$$

3) Time on Air range (ToA_{range}): it corresponds to the difference between ToA of the highest and lowest SF, i.e., 12 and 7, respectively, for a given percentage of compression (PC). ToA_{range} is calculated as follows:

$$ToA_{range_{PC}} = ToA_{SF_{max}} - ToA_{SF_{min}} \quad (10)$$

IV. Results and Discussion

Based on the methodology described in Section III-A, we calculate the percentages of compression for the resultant 64 rules that are aplied to LoRaWAN using the values listed in Table I. ToA is calculated based on Equations (2)–(8), and the results for each Spreading Factor are illustrated in Fig. 5. The evaluation of ΔToA and ToA_{range} are based on the compression percentages of the 64 rules illustrated in Fig. 5. Note that there are cases where several rules achieve the same percentage of compression; hence, fewer than 64 points are plotted for the curves of each Spreading Factor.

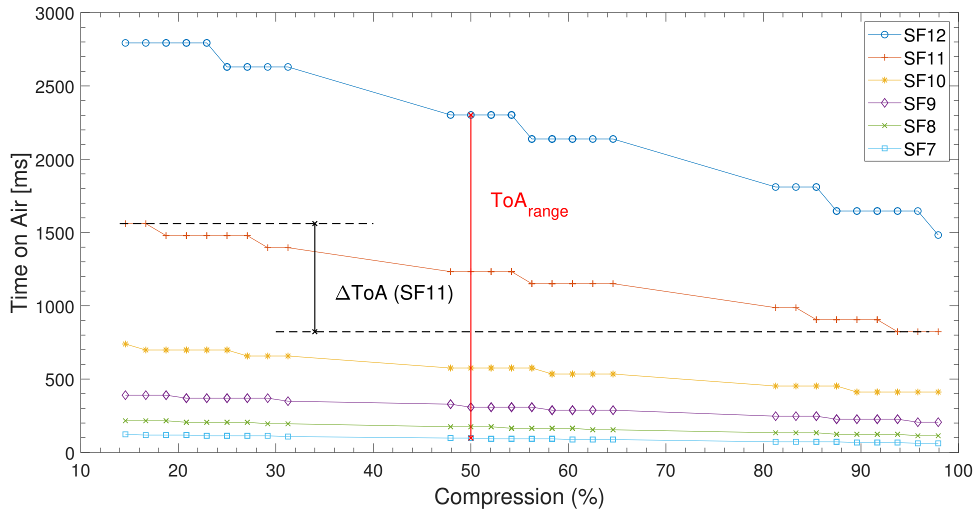


Fig. 5. Time on air according to the percentage of compression for different Spreading Factors

TABLE I
EVALUATION PARAMETERS

Parameter	Value
Spreadinging Factors	SF7 to SF12
Channel Bandwidth	125 KHz
Rule ID length	1 byte
UDP Payload	10 bytes
Explicit header mode (H)	disabled (0)
Adaptive Data Rate (DE)	1 (SF>=11), 0 (SF<11))
Cyclic Redundancy Check mode (CRC)	enabled (1)
Coding Rate mode (CR)	1

Fig. 5 shows that ToA increases when the percentage of compression decreases. This is expected since the percentage decrease results in larger SCHC packets. However, in Fig. 5 we observed that for each SF there are segments, called Constant Time on Air Segments (CTS), where ToA remains fairly constant despite the variations in compression percentages. This result is due to the step function *ceil* in Equation (7). For illustration purposes, we have also provided examples of the two additional metrics, ΔToA and ToA_{range} in the same figure. For instance, given SF11, ΔToA is 737.3 ms considering the PC_{min} and PC_{max} values. In the case of ToA_{range}, we illustrate the one for $PC = 58.3\%$, which is a percentage of compression common to 5 different rules. In this case, ToA_{range} is equal to 2045.7 ms.

Figure 6 shows the values of ΔToA for the different Spreading Factors. It should be noted that these values do not depend on the size of the UDP payload, since they only consider the percentages of headers compression achieved across all the set of rules. Knowing the behavior of ΔToA is useful for cases in which several packets from the same node require different compression rules; if the application is sensitive to delay variations, it is preferable to use smaller Spreading Factor (e.g., $SF7$) rather than a larger SF which entails larger delay variations across the different rules.

Fig. 7 shows the variation of ToA_{range} with the compression percentage. We observed that there are segments where the range does not vary despite the decrease or increase in compression percentages. This is due to the CTS identified in Fig. 5. ToA_{range} is useful for cases where the same compression rule is somehow fixed at the end device. In such a case, it is better to choose the highest compression percentage possible to avoid having a very high range in the transmission delay. Hence, if the LoRa link characteristics are modified, causing the packets to be transmitted at a lower speed (i.e., due to the modification of SF), the range of variation of ToA will be smaller, as it has been shown for the higher compression percentages.

Our analysis allows us to argue that the choice of the SCHC Rule should not only be based on the percentage of compression achieved by that specific rule. When evaluating the percentage of compression together with the behavior of ToA, we have been able to determine that there are ranges where the delays are insensitive to the compression percentages. In addition, we have also determined that there are compression rules that are much more sensitive to the variations of the Spreading Factor, resulting in an increase of ToA.

For example, suppose an end device is running an application that is sensitive to the variations of the delay (e.g., affecting timers or state machines) and the LoRa link is stable using $SF10$. When examining Fig. 5, it is possible to determine that ToA remains stable at 400 ms for rules that have a compression percentage between 99% and 89.7%; thus, these rules are the best candidates to be employed by the end device. Let us consider another example where an application has been set to use a fixed rule with a compression percentage of 93.75%. Fig. 7 shows that ToA_{range} for this compression percentage is 1579.8 ms. This value means that if the application starts operating with $SF7$, and given a change

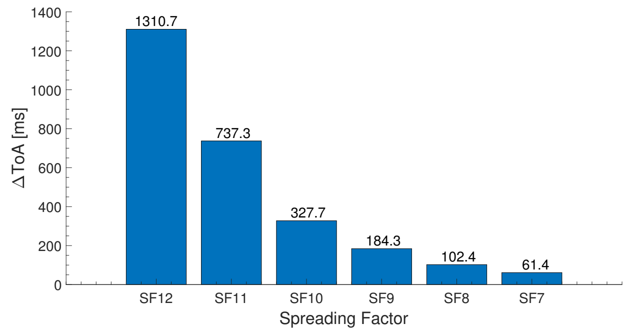


Fig. 6. Time on Air variation according to the Spreading Factor

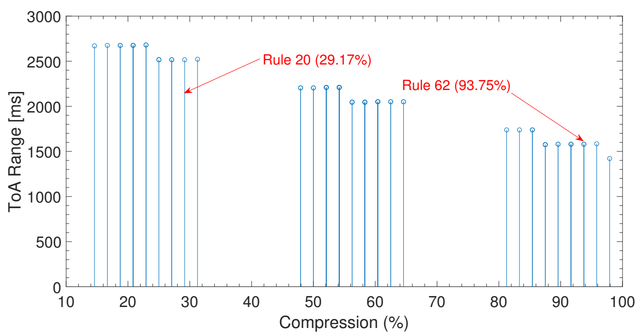


Fig. 7. Time on Air range according to the percentage of compression

in link conditions the device is forced to use $SF12$, the delay will increase at most by 1579.8 ms. Assume that the same application is set to use a rule with 29.17% compression percentage. If the application starts operating with $SF7$, and given a change in link conditions the device is forced to use $SF12$, the delay will increase at most by 2516.7 ms. The above indicates that when a variation of the Spreading Factor occurs, the second rule will increase its delay by almost 1 second more compared to the initial rule. This may have a great impact for applications that are sensitive to the delay introduced by the variations of the Spreading Factor.

V. Conclusions

In this work, we have conducted a study of the behavior of the transmission times, namely the *Time on air (ToA)*, under the Static Header Compression Scheme (SCHC) that is being standardized by IETF to enable IP-based communications in LoRaWAN. Two new metrics, ΔToA and ToA_{range}, have been proposed to analize the delay behaviour across different compression rules and different LoRa links (i.e., variations of Spreading Factors). Based on our study, we have concluded that the choice of the SCHC rule should not only be based on the percentage of compression achieved by the specific rule. Indeed, ToA does not have a linear behavior with respect to variations in compression percentages.

A segment called CTS (*Constant Time on air Segment*) is identified. Within this segment, ToA does not vary depending on the percentage of compression, which makes it possible to choose rules with lower compression percentages without affecting the delay introduced by ToA. The rules with lower compression percentages are useful when the packet has headers that cannot be compressed due to undetermined values (e.g., IPv6 addresses or UDP ports). Designing SCHC rules that are within the compression ranges that keep ToA constant may avoid a trade-off between versatility and an increased latency for packet reception in LoRaWAN.

References

[1] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of lorawan," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, Sep. 2017.

[2] W. Ayoub, M. Mroue, F. Nouvel, A. E. Samhat, and J. Prévotet, "Towards ip over lpwans technologies: Lorawan, dash7, nb-iot," in *2018 Sixth International Conference on Digital Information, Networking, and Wireless Communications (DINWC)*, April 2018, pp. 43–47.

[3] P. Weber, D. Jäckle, D. Rahusen, and A. Sikora, "Ipv6 over lorawan™," in *2016 3rd International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS)*, Sep. 2016, pp. 75–79.

[4] R. Sanchez-Iborra, J. Sánchez-Gómez, S. Perez, P. J. Fernández, J. Santa, J. L. Hernández-Ramos, and A. F. Skarmeta, "Internet access for lorawan devices considering security issues," in *2018 Global Internet of Things Summit (GIoTS)*, June 2018, pp. 1–6.

[5] D. Guinard, "Towards opportunistic applications in a web of things," in *2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, March 2010, pp. 863–864.

[6] A. Minaburo, L. Toutain, C. Gomez, D. Barthel, and J. Zuniga, "Static context header compression (schc) and fragmentation for lpwan, application to udp/ipv6," Working Draft, IETF Secretariat, Internet-Draft draft-ietf-lpwan-ipv6-static-context-hc-21, July 2019. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-lpwan-ipv6-static-context-hc-21.txt

[7] D. Singh, O. G. Aliu, and M. Kretschmer, "Lorawan evaluation for iot communications," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2018, pp. 163–171.

[8] Z. Sheng, S. Yang, Y. Yu, A. V. Vasilakos, J. A. McCann, and K. K. Leung, "A survey on the ietf protocol suite for the internet of things: Standards, challenges, and opportunities," *IEEE Wireless Communications*, vol. 20, no. 6, pp. 91–98, 2013.

[9] K. Q. Abdelfadeel, V. Cionca, and D. Pesch, "Dynamic context for static context header compression in lpwans," in *2018 14th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, June 2018, pp. 35–42.

[10] "OpenSCHC," 2019. [Online]. Available: https://github.com/openschc/openschc/wiki

[11] Semtech, "SX1272/73 - 860 MHz to 1020 MHz Low Power Long Range Transceiver Datasheet," Semtech, Document, January 2019, revision 4. [Online]. Available: https://www.semtech.com/uploads/documents/SX1272_DS_V4.pdf

[12] L. Alliance, "LoRaWAN® Specification v1.0.3," Lora Alliance, Document, 2018. [Online]. Available: https://lora-alliance.org/sites/default/files/2018-07/lorawan1.0.3.pdf