UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Contribution to the Publish/Subscribe Communication Model for the Development of Ubiquitous Services in Wireless Sensor Networks

By
Ernesto J. García Davis

Ph.D. Advisor:
Dra. Anna Calveras Augé

Thesis submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Telematics Engineering
in the
Department of Telematics Engineering

Barcelona, 2019

A mi hijo Ernesto para que nunca olvide que
con la actitud, fuerza, perseverancia y fe en Dios se logran las metas.

Agradecimientos

En primer lugar doy gracias a Dios quien siempre ha guiado mi camino y me ha permitido alcanzar las metas que me he propuesto.

Agradezco a mi Madre cuyos consejos y sabiduría me enseñaron a ser perseverante en mis objetivos. Hoy se reafirman sus palabras cuando me decía que debería ser Profesor o Doctor, sin saber que un día esta combinación se haría realidad.

A mi Tía y mi hermano, por darme sus buenas vibras y estar pendientes de mi progreso a pesar de la distancia.

Debo agradecer muy especialmente a esa persona que sacrificó todo y decidió emprender esta hazaña conmigo lejos de nuestras familias y de nuestro país, mi esposa Anabelle. Ella siempre ha sido mi amor, amiga, compañera y confidente quien fue mi soporte en las buenas y en las malas a lo largo de mi estancia en Barcelona, España.

Sin duda, agradezco a Dios por hacerme el regalo más hermoso durante mi estancia en Barcelona, España: mi hijo Ernesto, quien me ha dado la fuerza para nunca claudicar en este trabajo y enseñarle a Él que la actitud la fuerza y la perseverancia y la fe en Dios es la clave para lograr hasta los sueños más inalcanzables.

Un especial agradecimiento a todos mis compañeros durante mi estancia en Barcelona: August, Carolina, Felipe, Paola y Karen con los que compartí grandes momentos de alegría y también superamos juntos situaciones difíciles.

Sin duda, expreso mi más sincero agradecimiento a mi directora de tesis Anna Calveras por su apoyo incondicional y por inculcar en mi ese criterio crítico en el desarrollo de esta tesis.

Finalmente y no menos especial, agradezco a la Universidad Tecnológica de Panamá y a SENACYT/IFARHU por haber creído en mí y darme la oportunidad para estudiar este Doctorado en Ing. Telemática con el cual espero transferir los conocimientos adquiridos a estudiantes, docentes e investigadores de mi área, en beneficio del desarrollo de nuestro Panamá.

A todos gracias…..

Abstract

Due to advances in wireless technologies and rapid development in embedded electronics, wireless sensor networks (WSN) have become one of the key technologies of the Internet of Things (IoT). The WSN have the ability to measure physical phenomena of their environment, process and communicate this information using wireless technologies so they play a leading role in the development of applications that respond or adapt to the context of a user, as is the case of ubiquitous environments such as smart cities, industrial automation, e-health among others. Also, IoT trend has opened the possibility that smart objects or devices are also capable of exchanging status information, conditions and capacity with the purpose of interacting with each other, in the same way that traditionally human beings have done it through of systems based on presence. These systems require information from an event in real time to react in a timely manner to the conditions or the context of the user or device. These applications open new challenges in the management of WSN resources since these networks operate in environments that are generally prone to packet loss and they are composed of generally tiny nodes with limited resources in memory, processing, bandwidth and power.

Although Publish/Subscribe protocols in the WSN could become a solution to meet the requirements of these application, they must be adapted to the limitations of WSNs to help preserve scarce WSN resources such as bandwidth and energy. Also, it will be necessary to provide mechanisms that provide QoS in aspects such as: reliability and timeliness on the packet delivery corresponding to an event information.

The main objective of this thesis is the development of several mechanisms that allow for the adequacy of the Publish / Subscribe protocols to the features and limitations of the WSN for the provision of ubiquitous services in the context of the IoT. In addition, QoS support is supplied through mechanisms that provide reliability and timeliness in packet delivery and data aggregation techniques are applied to be efficient in the energy consumption and bandwidth of the WSN. Our research begins by identifying the need for a protocol considering key aspects such as reliability, sleeping support and messaging format for the provision of a presence service in the WSN. We propose an architecture to provide presence service for WSN based on a Publish / Subscribe model distributed and focused on mechanisms such as the aggregation of data and the publication of messages on demand to achieve efficiency on energy and bandwidth. All these mechanisms have been applied in the design of a system called PASH addressed to home control based on Ambient Assisting Living concept.

Reliability provided by Publish / Subscribe protocols on WSN become of importance in the design of applications that require to receive a message to react on time or in real-time to an event. We initially focus our study on increasing the packet delivery ratio (PDR) in the destination node through the improvement of reliability mechanisms. We evaluate the reliability mechanism of MQTT-SN protocol and several proposed mechanisms of CoAP protocol. From this evaluation, we propose a new and simple adaptive retransmission mechanism to respond to the loss of packets in the most appropriate manner.

Finally, we consider applications such as: e-health, critical infrastructure control and monitoring among others need to meet different QoS requirements such as reliability and timeliness for each type of message received. Also, data aggregation techniques play an important role in WSN to reduce energy consumption and bandwidth. In this thesis, we

propose a mechanism that provides to application three different QoS levels: we provide an improvement of our previous retransmission mechanism for reliability, we include data aggregation in our reliability mechanism and we provide timeliness mechanism on packet delivery.

Content

List of Tables

List of Figures

Glossary

WSN	Wireless Sensor Network
IoT	Internet of Things
XMPP	Extensible Messaging and Presence Protocol
XEP	XMPP Extension Protocol
SIP	Session Initiation Protocol
SIMPLE	Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions
IMP	Instant Messaging and Presence
HTTP	Hypertext Transfer Protocol
IPSO	Internet Protocol for Smart Object
IETF	Internet Engineering Task Force
IEEE	Institute of Electrical and Electronics Engineers
RFC	Request for Comments
W3C	World Wide Web Consortium
XML	Extensible Markup Language
BXML	Binary XML
EXI	Efficient XML Interchange
CoAP	Constrained Application Protocol
QoS	Quality of Service
MQTT-SN	Message Queuing Telemetry Transport – Sensor Network
REST	Representational State Transfer
URI	Universal Resource Identifier
IM	Instant Messaging
GPS	Global Positioning System
ASCII	American Standard Code for Information Interchange
CoCoA	CoAP Simple Congestion Control/Advanced
TCP	Transport Control Protocol
UDP	User Datagram Protocol
6LowPAN	IPv6 over Low-Power Wireless Personal Area Networks
RTO	Retransmission Timeout
RTT	Round-trip Time
CSMA-CA	Carrier Sense Multiple Access/with Collision Avoidance
ACK	Acknowledgement
PHY	Physical
MAC	Media Access Control
FFD	Full Function Device
RFD	Reduced Function Device
PAN	Personal Area Network
LR-WPAN	Low-Rate Wireless Personal Area Network
PDR	Packet Delivery Ratio
IMP	Instant Messaging and Presence
SRTT	Smoot RTT
PASH	Presence-aware Smart Home System

PST	Publish/Subscribe Tree
CIC	Central Intelligent Coordinator
AIC	Area Intelligent Coordinator
SDO	Smart Devices or Objects
RTTVAR	RTT Variance
DT	Deadline Target
EWMA	Exponential Weighted Moving Average
DM	Deadline Mechanism
DF	Deadline Factor
MDR	Message Delivery Ratio
PTR	Packet Timeliness Ratio
RTX	Retransmission
RPTXR	Retransmission Packet Ratio
RMTXR	Retransmission Message Ratio

1. Introduction

Advances in micro-electronics and wireless communications have allowed the rise of Wireless Sensor Networks (WSN). Due to the ability of these networks to capture a wide variety of context information, they have been applied in different areas such as: health [1], agriculture [2], transport [3], home automation [4], among others.

Moreover, the trend in the use of WSN has changed from being simply applied in the monitoring of events to being integrated into systems that respond or adapt to the context of a user, as is the case of ubiquitous environments such as building, home, and industrial [5] automation, smart cities [6-7], e-health [8]. Therefore, these systems require obtaining information from an event in real time to react in a timely manner to the conditions or the state of the user or device. An example of these, is the case of presence-based systems. The goal of these systems is the storage and distribution of information related to the availability of a person to establish a communication, which is known as presence information.

This information has been traditionally used by personal applications such as Instant Messaging (IM) [9] and it refers to communication conditions, capacity and preference of a user to know the willingness or responsiveness of other human users to engage a communication. In these systems, presence information has evolved from a simple "online / offline" binary state used in the beginning by instant messaging applications, to become more complex states that describe the characteristics of a user's context (space, time, services, resources, activity, among others).

Moreover, the idea of presence associated with a single source of presence (application) has been lost to adopt a much more ubiquitous concept where information can come from computers, mobile devices, applications and sensors of the user's environment.

In a presence-based system, entities that have associated presence information generate this information asynchronously through changes in their context.

Nowadays, smart object or devices could use this information to discover the status, conditions and capacity with the purpose of interacting with each other. In this thesis, a “smart object or device” refers to anything that has specific functionality and that is able to compute and communicate on its own.

In the case of sensor nodes, the traditional mechanisms to obtain information from the WSN are not the most adequate to comply with these systems that require information in real time. First, the mechanism in which the nodes periodically report the information of a detected event would not be useful for the presence-based system. In this method, the system would receive information on events that occurred in the past, therefore the response or action that was executed would be delayed. On the other hand, if the classic request / response method is used, the presence-based system would need to continuously query the wireless sensor network to obtain up-to-date information on a given event. This situation, results in an increase in network traffic and therefore a high risk of network congestion. Additionally, this would impact the life of the network [10], since the sensor nodes, usually operated by battery, would exhaust their energy faster due to greater use of their processing and communication resources.

A first solution to this problem requires that the queries be proactive, that means that they are stored in the WSN so that once a change of presence occurs, it can be notified to the

interested parties. Considering that presence changes are events that produce sources of presence, these systems fit perfectly with the Publish / Subscribe communication model [11].

In addition, due to the event-based nature of sensor networks, the integration of the Publish / Subscribe communication model in these very restricted environments has been a field of study in past years [12] - [15] but it remains as a hot topic for research community.

On the other hand, in the last years presence information was limited traditionally to be shared only among human beings. Nowadays, the tendencies towards new paradigms such as the "Internet of Things" (IoT) [16], open the possibility that the presence information can be exchanged between objects as well as between objects and individuals. In this way an intelligent environment is created in which objects can react depending on the presence information of other objects or individuals in their environment.

However, there were not standard protocols for presence service on WSN. In view of this situation, the most common option to implement presence service on WSN has been the adoption of existing standard protocols such as SIP/SIMPLE [17] and XMPP [18] on Internet providing this service.

Besides, the growing demand for applications that brings the Internet of Things such as: Supply chain management, monitoring and control of critical infrastructure, intelligent transport systems and intelligent security surveillance and many others, demand increasingly stringent requirements that allow these applications can be supported in these wireless sensor networks. This refers to provide to these applications the required quality of service [19] features such as: the reliability and timeliness on the packet delivery corresponding to an event information, and the efficient use of resources on WSN such as: energy and network bandwidth. Quality of service has become an emerging area research and a challenging task for protocols working on WSN due this kind of network is generally composed by nodes with limited resources in terms of processing, memory, energy and bandwidth. In addition, these networks operate in environments that are generally prone to packet loss and the nodes generally operate for a lifetime based on the source of power that has been provided to them.

These are research topics still with a lack of proposal and thus in this thesis we address these issues to define mechanism to provide and improve packet delivery, timeliness and use of efficient resources in WSN using publish/subscribe protocols.

1.1 Motivation

We consider that the evolution of the Internet of Things has brought as a consequence that the uses of WSN are moving from the mere monitoring of certain environmental variables towards its adoption for context aware systems. WSN are able to capture a rich set of contextual information such as: spatial, physiological and environmental data and to react regarding to a situation without user interaction. However, WSN also could obtain presence information. In this thesis, we initially focus on identifying the importance and usefulness to consider presence information on WSN to enhance communication and interaction among smart objects.

The presence protocols standardized on the Internet such as SIP/SIMPLE [17] and XMPP [18] are not suitable to be implemented directly in the WSN, because these networks have limitations in terms of processing capacity, bandwidth and energy. This results in the need for a protocol that makes efficient use of these resources to prolong the life of the network.

We have evaluated the features of both protocols and have identified that existing protocols for presence service could not be implemented directly in the WSN because they do not consider aspects which are specific characteristics of these types of networks, such as:

low bandwidth, limited resources on memory, processing and energy and lossy environment. Therefore, we consider important that the design of a presence service must consider the efficiency in the use of resources and the optimization of communication in the WSN.

The standard protocols for presence service are based on Publish/Subscribe communication model. In this context, presentities are publishers and watchers are subscribers that are interested in a presence event. Although these protocols could not be directly deployed on WSN, we could take advantage of some protocols that work on WSN with the publish / subscribe communication model in order to provide presence information in this kind of network.

In this sense, we consider there are some efforts such as CoAP [20] protocol developed by IETF to standardize a mechanism to transfer information from/to nodes on WSN. CoAP protocol is based on REST architecture thus uses request/response functionalities of HTTP protocol to communicate in a lightweight way with constrained devices such as sensor nodes. In addition, CoAP defines an Observe [21] model based on Publish/Subscribe communication to obtain information of an event when this is generated by these nodes. On the other hand, MQTT-SN [22] is a standard protocol by OASIS, which is based on Publish/Subscribe communication focused on WSN to obtain information from events in the near "real time". Both protocols allow the development to applications based on presence information on WSN.

As we mentioned, IoT vision will allow every smart object or device to be connected sharing presence information which would result in WSN with a number of nodes in the order of ten, hundreds even thousands of nodes. This is referred as scalability and is a critical issue to cope for application in WSN. In this situation, Publish/Subscribe model could provide a distributed architecture or centralized one. Both of them provide advantages and drawbacks that we analyzed and, in this thesis, proposed a new one to cope with the scalability challenge on WSN.

Although Publish/Subscribe could be the appropriate model to obtain information from events from sensor nodes, there are several requirements that becomes critical for applications. These requirements are related to quality of service parameters such as: reliability and timeliness on packet delivery because some message about critical information must be delivered to destination and other messages must be delivered on time to the destination. The most of protocols based on Publish/Subscribe on WSN lacks of mechanism to provide these features to application. In addition, application could require expressing the QoS feature depending on the type of message. These requirements could become of paramount importance in the design of applications that require to receive a message to react on time or in real-time to an event.

Most of the protocols in Publish / Subscribe in WSN do not provide these features or use very simple mechanisms that do not fully comply with the expected quality of service. In terms of reliability, the most of protocols such as: MQTT-SN and CoAP use reliability mechanisms in the delivery of packets based on a fixed packet retransmission mechanism. This is an important drawback because they do not consider the conditions of the network in which they operate to calculate an RTO (Retransmission Timeout). Considering this need, we initially evaluate both protocols and propose a new and simple adaptive retransmission mechanism considering parameters that represent the conditions of the network to respond to the loss of packets in the most appropriate manner.

Subsequently, CoCoA [23] proposed a dynamic retransmission mechanism for the CoAP protocol, which using two estimators in parallel provide more network information to

calculate an RTO adapted to network conditions. In this sense, in this thesis we include a new proposal of our previous adaptive retransmission mechanism and we evaluate and compare it with CoCoA and present its advantages and drawbacks.

In the case of timeliness, the protocols that implement this feature make use of the priority and do not consider network parameters such as RTT that allow knowing the conditions of the network at a given time. Sending packets to the destination in conditions where the network is congested initially causes a greater delay that will result in a late packet delivery at the destination. This results in the wasting of resources such as energy and bandwidth, which are very restricted in this kind of networks.

Based on that, we propose a timeliness mechanism that, based on the deadline established by the destination and in the conditions of the network, the decision is made to send the packet to ensure that the packet arrives at the destination according to the established timeliness requirement.

Additionally, as mentioned, the bandwidth in the WSN is a very limited resource and therefore the efficient use of it must be an aspect to consider for the development of applications in these types of networks. A mechanism such as data aggregation [24] would take advantage of the bandwidth capabilities of WSNs. This mechanism provides the ability to place more messages in a single packet that will be sent to the destination. This situation has been addressed in part of this thesis and in which it is proposed to make use of the aggregation of data at times when the loss of packets has been reported or detected. In this way, the risk of congesting the network and causing greater packet losses towards the destination is reduced.

The most of protocols in literature could provide at most two of the features of quality of service above mentioned and also there is a lack of mechanisms that allow applications to express or negotiate quality of service requirements so that they can function correctly in WSNs.

For this reason, in this thesis initially the presence technologies used in the Internet will be studied, the requirements to provide a WSN presence service will be defined and an architecture will be designed to offer this service using the resources of the WSN efficiently. And finally, mechanisms to provide a better reliable packet delivery, timeliness and data aggregation will be proposed.

1.2 Thesis Methodology

In this thesis, we focus our scope of research on communication of nodes (objects or devices) inside WSN. The reason of that is because trends such as Internet of Things and presence service in WSN considers nodes can exchange event information among them and not only with the others located on external networks.

Our main objective is the development of several mechanisms that allow for the adequacy of the Publish / Subscribe protocols to the features and limitations of the WSN for the provision of ubiquitous services in the context of the IoT. On the other hand, the effect of the routing protocol is beyond the scope of the study of this thesis, since we focus especially on the performance of the publish/subscribe protocols analysed.

Considering the scope depicted above, the main research question of this thesis is: How to achieve event information communication for ubiquitous applications in the WSN ?

To achieve this objective, we established several tasks and described the objective of each one, they are the following:

1. Provisioning of Presence Service using Publish/Subscribe protocols in WSN

In this task, we focus on identifying the importance and usefulness of exchanging of presence information to enhance communication among smart objects and to react properly to their environment.

The objectives of this task are:

- To evaluate protocols providing presence service on Internet mainly focusing our attention to the fact that they should be used over wireless environments and resources-constrained devices such as sensor nodes.
- To propose several design points to consider for a presence service protocol on WSN.
- To evaluate Publish/Subscribe protocols on WSN focusing on important design aspects that impact the resources of WSN such as: reliability, sleeping support and messaging format to save energy.
- To design and evaluate several mechanisms to provide presence service on WSN using Publish/Subscribe protocols considering efficiency in the use of resources and the enhancement of communication.

2. Improving of Packet Delivery Ratio on Publish/Subscribe Protocols

In this task, we evaluate the reliability mechanism of MQTT-SN protocol and the CoAP protocol. For MQTT-SN we only consider its QoS level 0 (non-persistent mode) and QoS level 1 (persistent mode). We do not consider QoS level 2 since it requires that more messages be exchanged between source and destination, and our objective is to optimize communication using the least number of messages. For CoAP protocol, at this time we consider its default congestion mechanism.

The objectives of this task are:

1. To evaluate the reliability of the publish/subscribe protocols on WSN to deliver a packet.
2. To propose and evaluate a reliability mechanism with a new dynamic retransmission scheme that considers network conditions based on RTT value to compute a suitable RTO.

3. Provisioning of QoS Support for Publish/Subscribe Protocols

In addition to the reliability in the delivery of packets that require applications to have the ability to respond appropriately to an event, another aspect to consider is the timeliness that refers to the guarantee of on-time delivery of packets. These are important aspects to consider for the publish/subscribe protocols in the WSN. Thus, in this task we focus on design and evaluate several mechanisms to provide these additional features of quality of service and to express or negotiate quality of service requirements so that applications can meet their requirements in WSNs.

The objectives of this task are:

1. To design and evaluate a mechanism that establishes three different QoS levels based on Publish/Subscribe model for wireless sensor networks.
2. To enhance our previous adaptive RTO mechanism by dynamically adjusting the K value parameter.
3. To include the use of data aggregation technique to reduce congested situations and to improve performance.
4. To design and evaluate a priority mechanism to provide timeliness for CoAP Protocol
5. To design and evaluate a deadline mechanism to provide an enhancement timeliness feature.

1.3 Organization

This thesis is organized in six chapters. The current chapter gives the introduction and motivations into the investigation to carry out this thesis. Chapter 2 covers the state of the art. Chapters 3 to 5 represent the contribution of this thesis. Chapter 6 presents the conclusion of this thesis and provides ideas for future work. In the following, the details of the content presented in each of the chapters is given.

Chapter 2 provides an overview and state of the art of presence service, WSN and also Publish/Subscribe Model. We focus on protocols and the features related to these topics addressed in this thesis.

An evaluation of protocols for presence service on Internet is presented on Chapter 3. Also, it introduces the requirements to implement presence service on WSN. In addition, an architecture based on Publish/Subscribe providing presence service on WSN is defined and finally we present a system approach proposal of an Ambient Assisted Living system based on this architecture to provide safety, security and comfort at home that we have called PASH.

In Chapter 4, we evaluate the reliability to deliver a packet to provide quality of service of the publish/subscribe protocols on WSN. The default congestion mechanism of CoAP protocol and MQTT-SN were compared based on its retransmission scheme. Both of them, are based on fixed retransmission scheme, we proposed a new dynamic retransmission scheme that considers network conditions based on RTT value to compute a suitable RTO. This new mechanism improved the packet delivery ratio (PDR) on the destination compared with the default congestion mechanism of CoAP protocol and MQTT-SN results.

In chapter 5, we enhance our previous proposed mechanism on chapter 4 which adjusts the RTO depending on the subscriber PDR. This new technique is used to propose a mechanism that establishes different QoS levels based on Publish/Subscribe model for wireless sensor networks is introduced. This mechanism mainly provides reliable delivery of packet and timeliness to meet application requirements. Also, it provides data aggregation to be efficient in terms of energy consumption and the use of network bandwidth. We evaluate our proposal and we compared it with the new CoCoA mechanism of CoAP protocol and MQTT-SN.

Finally, conclusions, publications generated from this thesis and some future work guidelines are exposed in Chapter 6.

2. State of the Art

In this chapter, we firstly begin with a general vision on presence notification service. We present related works on WSN integration with presence service. Followed by this, Publish/Subscribe model is explained as well as related works on publish/subscribe protocols on WSN since it is the basis to integrate WSN on several applications where information is required on real-time to react adequately. Finally, quality of service to meet the requirements of applications to work properly in WSN is presented. In addition, we present the related works on publish/subscribe protocols in WSN providing features of quality of service such as: packet reliability, data aggregation and timeliness.

2.1 Presence Notification Service

After several efforts by the IETF to define a protocol and a standard data format for the presence service on Internet, it was only possible to define a model for the presence service called IMPP [9] such as showed in Figure 1. This model establishes the entities involved and the services that an IMP (Instant Messaging and Presence) system must offer.

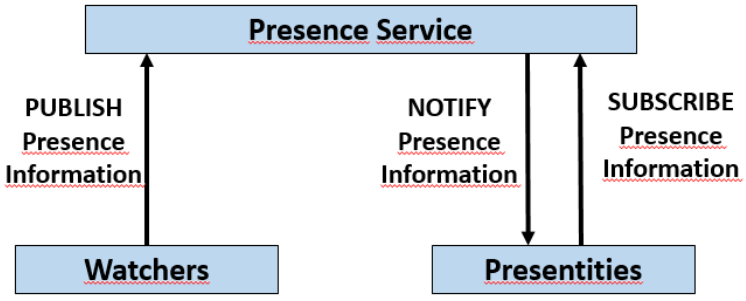


Figure 1 IMPP Presence Model Entities

An IMP system has two types of entities: "presentities" and "watchers". A "presentity" is an entity that has an associated presence information and provides information to interested parties through the IMP system. A "watcher" is an entity interested in the presence information of a "presentity" and obtains it through the IMP system. The presence service accepts, stores, and distributes the presence information to everyone subscribed to get the notification about presence changes.

In addition, IMPP provides the semantics [25] and the standard format [26] of information to be used among IMP systems.

As we mentioned, thanks to the ability of the WSN to obtain information on environmental parameters, these networks are being integrated into presence notification systems. On these systems the information from real-time events is required to respond or react in a timely manner to the conditions or the state of the user or device.

In the beginnings of presence-based systems, presence information represented simple states [26] such as: "online, offline", however the need for more information to be able to interact with an individual or devices brought with it the need for new formats that include

information on their mood and activities [26], additional contact information of a person [27] and also information on their geographical location [28]. However, most of these presence information formats are not suitable for integrating the presence information provided by the sensor nodes. A first alternative to adapt the presence information was presented in [29] and [30] where the authors propose the addition of two new attributes to the GEOPRIV format [28]. However, there was no consensus in the data model used to describe the presence information of objects and devices. In fact, there was a lack of standardization in the data and metadata exchange to achieve an interoperability between devices and applications. Nowadays, we consider the proposed Smart Object Model by IPSO Alliance is a real effort to solve this gap. This model initially provides to CoAP protocol with a common set of object definitions and pattern designs that enables interaction between any application with any devices.

As we mentioned, there are not standard protocols for presence service on WSN. The most common option to implement presence service on WSN has been the adoption of existing standard protocols such as SIP/SIMPLE (Session Initiation Protocol/Session Initiation Protocol for Instant Messaging and Presence Leveraging) [17] and XMPP (eXtensible Messaging and Presence Protocol) [18] on Internet providing this service.

In SIP/SIMPLE, it establishes an event notification framework to use the Publish / Subscribe model, and the SIMPLE extension presents several centralized elements to integrate the Publish / Subscribe model into presence-based systems. These centralized elements provide the means to contact another device in the network. One of the most important drawbacks to deploy this protocol in WSN is the large message size. Each of these messages contains ASCII characters which each occupy a byte of space. This results in extensive messages to fit the size of messages handled by the sensors (127 bytes according to the IEEE 802.15.4 standard [31]).

On the other hand, XMPP [18], is a standardized and well-established technology for real-time communication. Specifically, its extensions XEP-0060 [32] and XEP-0163 [33] define a publish/subscribe framework and a set of elements to express presence information through this framework. In the same manner as SIP / SIMPLE, XMPP introduces a centralized element, in which the presence information of the users is managed. In this sense it presents the same problems as SIP / SIMPLE we mentioned above. In addition, XMPP messages are based on XML (eXtensible Markup Language). These messages require a processing capacity to analyze and interpret them, which results in additional complexity for the limited resources possessed by the sensor nodes [34]. There are several proposals that seek to simplify the processing and transmission of messages encoded in XML. These are based on the efficient compression of XML [35]. The W3C (World Wide Web Consortium) has defined EXI [36] (Efficient XML Interchange) and in [37] BXML (Binary XML) is presented. However, the main disadvantage of this class of compressed formats is that they may require more resources, in terms of memory and CPU, than the processing of the XML file itself [34]. An additional disadvantage that XMPP presents with respect to SIP / SIMPLE is that it only works on the TCP protocol. TCP's connection establishment, congestion control, and end-to-end flow control mechanisms could lead to low performance in wireless environments and a greater impact on power consumption in these types of networks [38].

Several efforts to cope with these problems are presented in [39], [40] and [41] where they propose several optimizations and improvements on XMPP protocol for implementing publish/subscribe scheme, sleeping nodes support and UDP transmission support features on constrained resources sensor nodes to save energy in this kind of network. However, none of

these works consider Quality of Service support to provide reliable delivery of packet to destination which is of paramount importance for application working on WSN.

The major part of works proposes a gateway-based solution where a central node is between WSN and Internet. In addition, this node is in charge of communication with WSN and translating or converting messages to be compatible between these networks [42-45]. A multi-protocol proxy architecture is provided in [42] which allows translations between messaging protocols such as XMPP, CoAP and MQTT. In [43] authors propose the implementation of a gateway that collects data from sensors nodes and converts it in appropriate XMPP format. In TinySIP [44] is proposed to use a compact and efficient message of the SIP / SIMPLE protocol for energy consumption saving to address the limited resources that WSNs possess. It limits the size of useful data (payload) of application to 29 bytes to achieve this objective. Similarly, TinyREST [45] uses the HTTP protocol based on the REST architecture [46] as the standard mechanism for accessing the WSN data. These proposals only focus on reducing message size to achieve energy efficiency, however, they do not consider that WSNs applications could require, in certain situations, reliability mechanisms because the data transmission is done through a wireless medium and it is usually multi-hop. This implies that in each hop the network conditions (load, congestion, data loss) are different, which increases the probability of packet loss. In [47] the authors propose the use of an application type called "bot" that works between the WSN and the presence service. The users in the external network are notified about the changes in the events that occurred in the WSN. Also, there are some works that propose integration of WSN with other type of networks such as cellular networks. In [48] and [49] an architecture is proposed for the integration of WSNs with 3G networks through the IP Multimedia Subsystem (IMS) to provide new services to users of mobile networks. The integration is carried out through a Gateway that is connected between the presence service located in the IMS network infrastructure and the WSN. In the Gateway, the messages received from the WSN are also transformed into standard SIP/SIMPLE messages to be compatible by the presence service in the IMS.

All the proposals analyzed up to this point consider that the sensor nodes are only information providers of events from the WSN to the interested parties that are in an external network, such as the Internet, cellular network or any other type. However, the situation is different under paradigms such as the IoT that we have already mentioned, where traffic can remain inside the WSN since an object may be interested in the presence information of another object. Considering the limitation of protocols explained above and the constrained resources of WSN such: network bandwidth, processing and energy, the attention has shifted to define new mechanisms to provide presence information inside WSN.

In the first part of this thesis, we study the presence technologies used in the Internet and we define the requirements to provide a WSN presence service.

The presence service is based on Publish/Subscribe Model. This model has proven to be an efficient method to obtain information from events from sensor nodes. In the next section, we present a general vision of Publish/Subscribe model, including its features and architecture that we could use to provide presence service inside WSN. Also related works on protocols in WSN benefiting from Publish/Subscribe model are presented.

2.2 Publish / Subscribe Model

The Publish / Subscribe communication model allows the interested party to subscribe to future events that may occur in a system. Therefore, the queries are stored in advance or

proactively so that they can be delivered in time to interested parties once an event of interest occurs. This benefits applications that require notification of real-time events, such as presence-based systems discussed in the previous section.

This model is composed by 3 entities, such as showed in Figure 2.

- Publishers: these are entities able to generate information commonly called au event and which is published to the notification service.

- Subscribers: are entities that are interested on the events or event patterns produced by publishers. They have the ability to express their interest on particular events by means of a subscription and are notified later by the notification service when an event matches their registered interest.

- Notification Service: is the infrastructure that serves as an intermediate point between publishers and subscribers. It is responsible for processing event publications, registering subscriptions and distributing event information to subscribers, which is called commonly notification. When a new event is generated and published to the notification service, it is responsible for checking the subscriptions that match the event to deliver it to the corresponding subscribers.

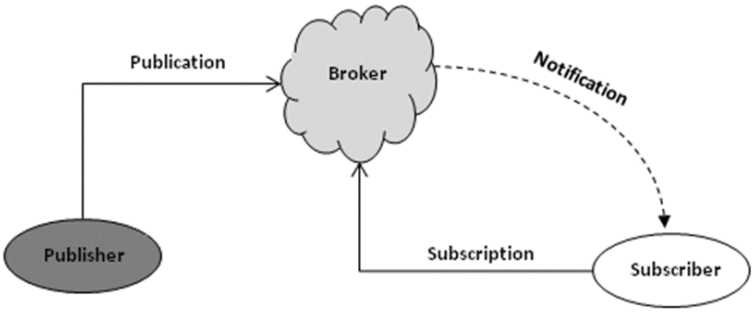


Figure 2 Publish/Subscribe Communication Model

The notification service provides the interaction between publishers and subscribers. This achieves that both publishers and subscribers can act independently of each other. This way of functioning allows the decoupling between publishers and subscribers in two aspects [11].

- Time decoupling: means that none of the entities (publishers or subscribers) need to be connected to the notification service at the same time when the publisher publishes an event. The publisher can publish an event while the subscriber is disconnected, and vice versa, the subscriber can receive notification of an event while the publisher that produced it is disconnected.

- Decoupling in space: this prevents any of the entities (publishers or subscribers) from having the need to know about the others. This means that publishers can post events directly to the notification service without the need to specify to whom the publication is intended. In addition, subscribers do not need to know which or

how many publishers produce the information of an event. In the same way publishers do not need to know who or how many subscribers will receive the information of an event.

Due to the high degree of decoupling offered by the Publish / Subscribe model and the event-based nature of sensor networks, the integration of this model in these very restricted environments has remained one of the fields of study for the scientific and research community [51 - 55].

There are some challenges for application development on WSN that could be solved applying the Publish/Subscribe communication model.

For instance, in application for real time monitoring purposes (fire detection or security surveillance) we need to know for future events and in the instant, it has occurred. Generally, applications submitted queries to sensor nodes each certain interval of time, however this scheme is useless in this case because we would only obtain past events and maybe no change has occurred in this interval of time.

If we apply publish-subscribe communication model, we can submit a query to the network in advance waiting for notification of the events that match the query in the future. In this case, the sensor nodes submitting the query are called subscriber nodes and sensor nodes that notify event have occurred are called publisher nodes.

On the other hand, in traditional schemes, applications interacting with nodes require to know the node address. However, nodes in WSN may change their network addresses at any time. Furthermore, sensor nodes could also fail and to be replaced by new ones. Therefore, this traditional way to communicate would result in communication failures.

With Publish/Subscribe communication model, it would not need to know the addresses of nodes, since information is delivered to the receivers (subscribers) not based on their hardware or network addresses but rather as a function of their content and interests. This scheme is called data-centric addressing.

Other advantages the Publish/Subscribe communication model offers are:

- The interacting parties (publishers and subscribers) do not need to be active at the same time. In particular, the publisher might publish some events while the subscriber is disconnected, and conversely, the subscriber might get notified about the occurrence of some event while the original publisher of the event is disconnected. This is compatible with nodes in WSN, because nodes at the most of time are idle in order to save energy.

- Publisher and subscriber nodes do not need to know the size of the network. It is compatible with dynamic network topology of WSN. Furthermore, synchronization is not required among publishers and subscribers. It allows for great scalability.

The presence service architecture fits into publish-subscribe communication model. Presentities and watchers communicate by exchanging presence events. Presentities are publishers in publish-subscribe communication model and watchers are subscribers that are interested in presence event.

It should be noted that presence service makes use of topic based publish/subscribe communication mode, because the watchers need to know every change in presence information of presentities.

However, for WSN where sensor nodes change its status from awake to sleep mode and vice versa, it could generate a lot of presence information resulting in higher bandwidth consumption considering 250 Kbps as the maximum bandwidth available on WSN based on IEEE 802.15.4 standard [9].

On the other hand, this model can be classified into two large groups: topic-based and content-based. In topic-based, events are categorized based on a set of topics. Subscribers will receive a publication about any type of change that happens in the event. Content-based ones allow you to express exactly what information you want to receive from an event.

The topic-based design can generate a lot of information which results in an additional increase in traffic, which would impact the resources of the sensor nodes. On the other hand, the content-based design requires more processing because the information in the message has to be extracted and interpreted in order to make the transmission decision. There is a compromise between the degree of detail of the information to be received and its impact on the resources of the nodes.

In the following section, the network architectures of the Publish/Subscribe communication model are presented and discussed. These network architectures mainly affect the degree of scalability of the system.

2.2.1 Network Architecture

Depending on the degree of scalability and simplicity that is required in the system, the Publish / Subscribe communication model can be implemented mainly in two ways.

2.2.1.1 Centralized Model

This model is represented in Figure 3, a central node that functions as an intermediate point of communication between subscribers and publishers, which is called Broker. This node is responsible for implementing all the functions of the notification service, explained above. That is, process the publications, register the subscriptions and make the event notifications to the corresponding subscribers.

This approach maintains the simple system design because all processing and control is done at the central point (broker). There are several protocols that make use of this model [15], [36], [44], [55]. In these protocols, the "sink" node acts as the central node of the architecture and also functions as the gateway to interconnect the WSN with external networks.

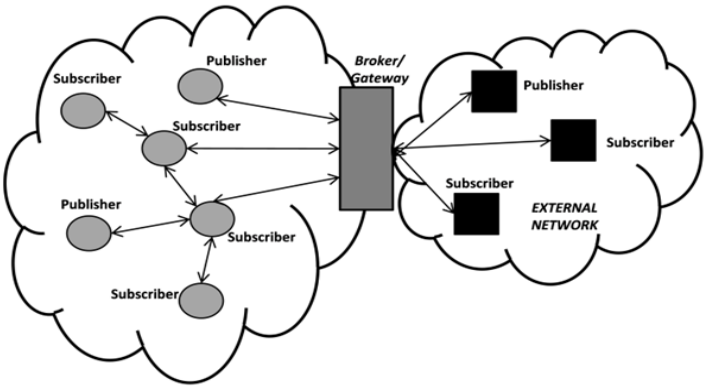


Figure 3. Centralized Publish/Subscribe Architecture

However, the centralized model potentially reduces the scalability of the system and introduces a single point of failure (Broker). This situation would reduce the stability of the system as well as being a potential bottleneck in situations of high load on the network, which would also affect the performance of the system.

2.2.1.2 Distributed Model

A solution to the previous model is to use a distributed model [56] as shown in Figure 4. In [57] a set of broker nodes connected by an overlay network is proposed. In this case, the functions of the notification service are distributed among the set of Broker nodes. This model allows the collection of subscriptions from subscribers and the routing of publications. The purpose of this model is to reduce the network load and increase the scalability of the system.

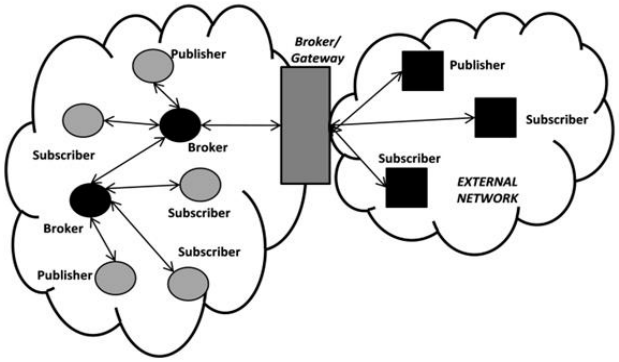


Figure 4 Distributed Publish/Subscribe Architecture

In addition, it is important to present the general operation of the Publish / Subscribe protocol. Figure 5 shows that the Publish/Subscribe service mechanism is located between Application layer and the rest of the other layers in the protocol stack for publisher node, broker node and subscriber node. On the publisher node, the Application layer is responsible of generating event packets that are transmitted to the broker node, where Publish/Subscribe protocol is in charge of verifying subscriber lists and performing QoS mechanism in order to satisfy subscriber nodes requirements. Finally, Application layer on Subscriber nodes receives event packets.

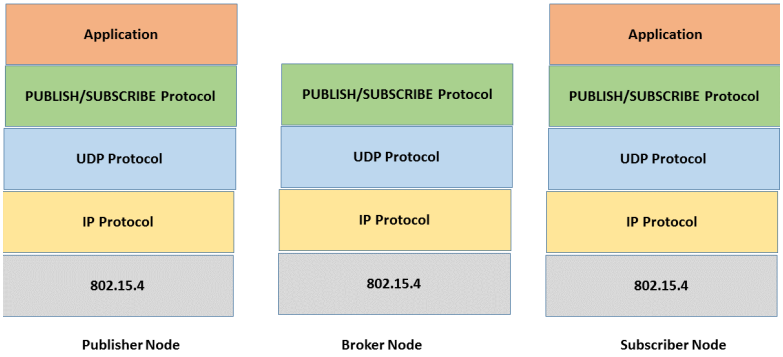


Figure 5 Publish/Subscribe protocol stack

The transport protocol mainly used in WSN is UDP since there are three fundamental problems with TCP over constrained-node networks. First, TCP has an increased header size compared with that of UDP (this issue is greater in 6LoWPAN networks, where header compression reduces UDP header even further, but no header compression has been defined for TCP). Secondly, TCP is not compatible with multicast, while many use cases in constrained-node networks involve multicast-type traffic (e.g. a user turning on a group of lights in a building). Finally, a TCP receiver provides feedback to a TCP sender regardless of the application on top. However, some applications in constrained-node networks may not require end-to-end reliability but might benefit from saving energy and bandwidth if reliability is not used. Despite these limitations, today there is a growing need to use TCP instead of UDP to achieve the integration of WSN protocols and IoT devices with the enterprise network infrastructures in which there are some limitations especially connectivity through the corporate firewall. Nowadays, research community is working on adapting TCP mechanism [50] such as windows size, RTO algorithm, long-lived TCP connections among others to implement TCP in IoT scenarios.

2.2.2 Publication Models.

Publications in the Publish / Subscribe communication model are generated by entities called publishers. There are mainly two publication models:

2.2.2.1 Explicit Publication Model

In this model, used in [15] and [22], publishers generate information and transmit the publications to the Publish / Subscribe infrastructure without considering that there are subscriptions related to the publication, as shown in Figure 6 Publications that do not have associated subscriptions become unnecessary traffic that affects the consumption of bandwidth, power and processing of the nodes of the network.

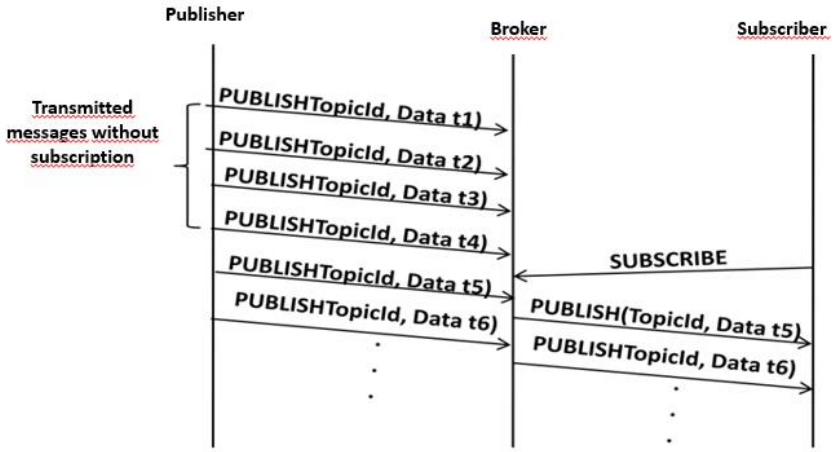


Figure 6 Explicit Publication Model

2.2.2.2 Publication on Demand Model

Unlike the previous model, in this case publishers only publish events if there is a subscription to the event in the system, such as showed in Figure 7. This model is used in [35], [57], [58]. The advantage of this model is that no additional messages are transmitted and therefore there is less impact on the network load. However, for this mechanism to work, publishers must be notified with some kind of mechanism, to indicate when they should start transmitting publications and in the same way when they should stop the transmission of publications.

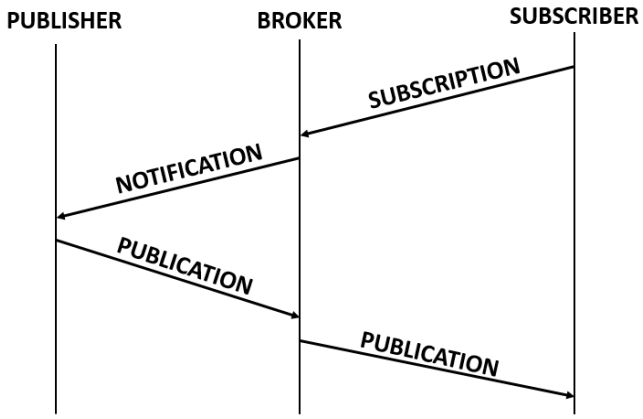


Figure 7 Publication on demand Model

2.2.3 Subscription Model

In the Publish/Subscribe communication model, subscriptions can be issued or canceled dynamically to reflect the change of interest in the information that subscriber wants to receive. There are two techniques for managing subscriptions, which impact on network traffic and information management by the Publish / Subscribe infrastructure and in particular the update of the content of the subscription tables.

2.2.3.1 Explicit Subscription Model

This technique is used by the majority of proposals such as: [22], [55], [59]. The transmission of a message (SUBSCRIBE) to the Publish / Subscribe infrastructure is required to reflect the interest of receiving information about an event. Once the subscription is made, it remains active until it is canceled as shown in Figure 8. This technique requires that subscribers who do not wish to receive more information about the subscribed event must send a message of cancellation of the subscription (UNSUBSCRIBE) to avoid receiving unnecessary further publications.

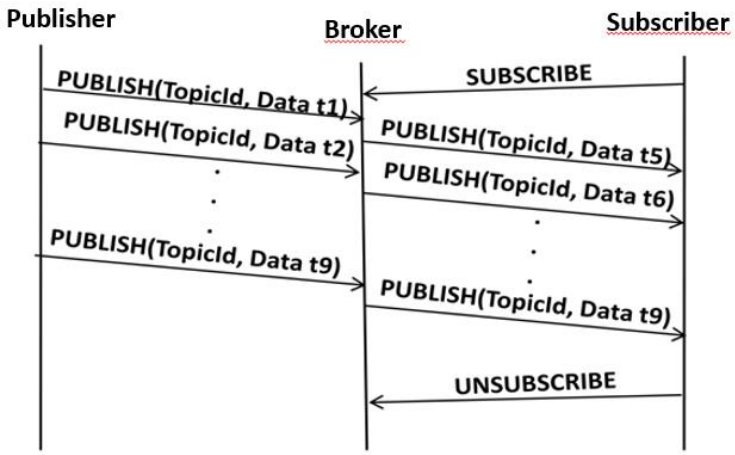


Figure 8 Explicit Subscription Model

One disadvantage of this technique is the case in which a node disappears from the network (for example by exhaustion of its power), as in the case of sensor nodes, the subscription will never be canceled and therefore publications will be sent unnecessary to the network.

2.2.3.2 Temporal Subscription Model

In this technique [44], [59] the nodes subscribe to an event with a specified subscription time, which is illustrated in Figure 9. If the Broker node does not receive a new subscription message to the event before the subscription time expires, the subscription is canceled. Figure 10 details the operation of this model. The subscriber node makes a subscription with a subscription time t3. From t0 to t3 the subscriber node receives the corresponding publications associated with its subscription and sends again a SUBSCRIBE message that updates the subscription time, this time it will be t5. However, because the Broker node does not receive a new SUBSCRIBE message before t5, the subscription is automatically canceled.

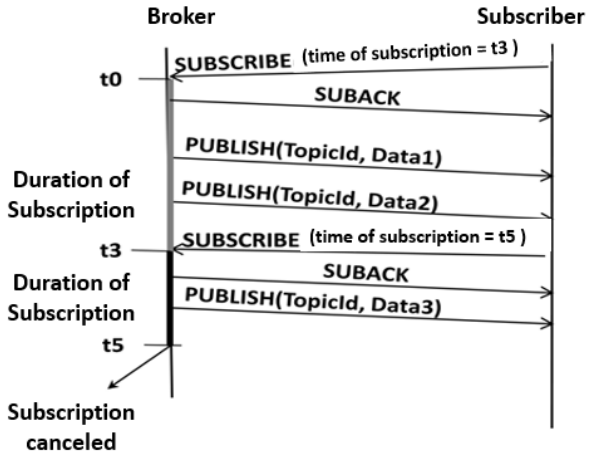


Figure 9 Temporal Subscription Model

This model would solve the disadvantage of the explicit subscription model, explained above. If a node disappears from the network, the subscription would be canceled when the subscription time expires, thus avoiding the transmission of unnecessary publications to the network. However, the subscription time is a design parameter to be considered, it can increase the network traffic and therefore the energy consumption of the nodes

There is another technique proposed in [20] in which every time a subscriber node receives a publication, its interest in remaining subscribed to the event is sent in the confirmation message. This technique is conditional on the publications requiring the subscriber to send a confirmation message, however there is a possibility, depending on the application, in which it is not necessary to receive confirmation of the publication from the subscriber node. For example, confirmation should only be sent if the published event refers to an alarm situation.

2.2.4 Messages Dissemination techniques on Publish/Subscribe model

One of the most important aspects of Publish / Subscribe systems is the use of the message dissemination mechanism that allow the messages of both subscriptions and publications to flow properly through the Publish / Subscribe infrastructure. There are several proposals in this regard.

2.2.4.1 Based on Unicast

This mechanism is used in the centralized network architecture of the Publish / Subscribe model. The nodes that wish to make a subscription send a unicast message to the corresponding broker. This mechanism is used in [22] where the broker goes through a list of subscribers and sends each one in unicast the publication of the subscribed event. This method could become inefficient as the number of nodes in the network increases.

2.2.4.2 Based on Broadcast

In this mechanism messages are sent to all nodes in the network to express interest in an event. In [58] and [60] a subscriber sends a subscription message to an event in broadcast mode. Each node receiving this message creates an entry in its routing table pointing to the neighbor node from which it received the subscription message. When a publication occurs, this path can be traveled to reach the subscriber of the corresponding publication. A variation of this proposal is presented in [61] in which only broadcast of publications is done. Subscriptions are propagated along the reverse path of the publications.

In [62] the construction of a broadcast tree called PST (Publish / Subscribe Tree) is proposed, which covers all the nodes in which the publications will be disseminated from a publisher to all subscribers. This proposal is only efficient when the number of publishers is small.

A different alternative is presented in [63]. This is based on a hierarchical structure to organize the network into multiple groups (clusters). Each cluster forms a subnet that uses the Publish / Subscribe model and has a representative node that acts as a broker. This node is responsible for the dissemination of subscriptions or publications between clusters. Although this proposal can potentially reduce traffic on the network, its complexity lies in the maintenance of its hierarchical structure under dynamic network conditions, as in WSN.

Broadcast-based dissemination is costly in networks where there may be a large number of publications and subscriptions generated by a large number of publisher nodes and potential subscribers such as WSNs.

2.2.4.3 Based on Gossip

Another alternative is to use a "gossip" form (epidemic) to disseminate subscriptions and publications. These mechanisms have in common that a received message is transmitted with a certain probability P <1. There are also certain variants of this mechanism. In some proposals, a randomness criterion is applied not only to decide whether to transmit or not, but also to select a subset of neighboring nodes to which to transmit the message.

A simple approach is presented in [64] where a random path is chosen to disseminate the information for each node. Each subscription is replicated in all visited nodes starting from the subscriber node. In the same way it happens with the publication to find a point where to find the subscription, as shown in Figure 10. If we consider that the random paths are long enough, it is highly probable that a publication will find or match all the subscriptions related to it.

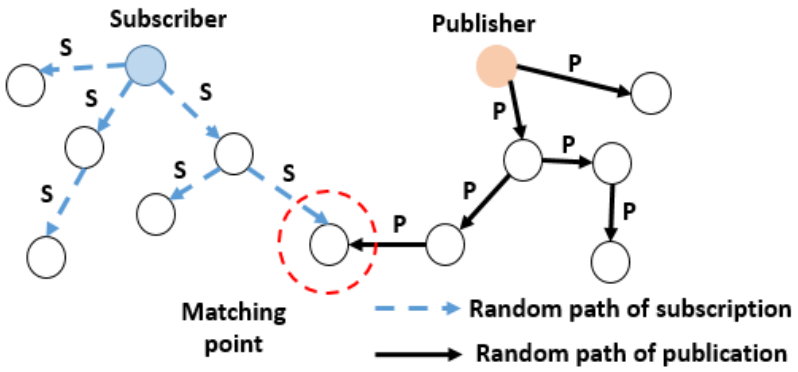


Figure 10 Gossip messages dissemination

In [65], it is proposed to use multiple random paths in order to shorten the notification time of the publication to the subscribers.

Another alternative is presented in [59] in which both subscriptions and publications are broadcast selectively. A subscription is broadcasted in broadcast based on a limit φ called subscription horizon. This defines the number of times a subscription is retransmitted in broadcast. On the other hand, for the propagation of a publication, only a fraction of the neighboring links in each node is selected.

However, gossip-based mechanisms incur high communication, storage and computing costs because a subscription or publication needs to be disseminated to many nodes in order to have the opportunity to find a point in the network where publications and subscriptions coincide.

Considering the most important aspects of Publish/Subscribe communication model above mentioned, in part of this thesis we evaluate and propose a distributed architecture based on Publish/Subscribe Model that allows distributing the load in the network by

grouping publisher and subscriber nodes in different broker nodes to offer presence service using the resources of the WSN efficiently.

2.3 Reliability on Packet Delivery

WSN transmits data through a wireless medium and is usually multi-hop. This implies that in each hop the network conditions (channel occupation, congestion, link quality) are different, which increases the probability of packet loss. A common approach to provide reliable delivery of packets is to use the acknowledgments (ACK) and the lost packet retransmission. Most of works propose a fixed retransmission scheme [12], [20], [22], [66-70]. Thus, they do not consider the network conditions that are crucial in order to calculate an appropriated RTO value. A short RTO value could give rise to spurious retransmissions, which leads to a waste of bandwidth, energy and computation, all of which are constrained resources in WSN. In addition, a larger RTO value may lead to a slow or late response to the loss of packets, which in turn results in an increase in the perceived delay and a decrease in the packet delivery ratio (PDR) at the subscriber node.

On the other hand, CoAP [20] is a standard application protocol suitable for WSN published on RFC 7252. CoAP has proposed a new mechanism called CoCoA (CoAP Simple Congestion Control/Advanced) [23]. CoCoA consider the RTT (Round Trip Time) as a measure of network condition to adjust the RTO value in a dynamic manner. One of the benefits of CoCoA is the use of two estimators in parallel that provide more network information to calculate an RTO value nearer to network conditions leading to obtain a more PDR by subscriber node. CoCoA potentially improves congestion control mechanism, but it could result in a more complex mechanism for constrained devices in WSN.

One of the contributions of this thesis is the evaluation of packet delivery reliability of publish/subscribe protocols on WSN. In this work we compare MQTT-SN and CoAP and we propose a new mechanism to compute RTO in a dynamic way considering network conditions. To the best our knowledge only the work on [23] considered the design of mechanism to compute a dynamic RTO considering network conditions such as RTT. However, this proposal is subsequently to the conclusion of our work.

2.4 Data Aggregation

It is common the data generated from different sensor nodes located in the same coverage area of an event are redundant. This redundancy can be exploited to achieve energy savings in transmissions, through appropriate filtering and data aggregation mechanisms [71-74]. There are a few works about data aggregation in publish/subscribe protocols. [75] proposes a data aggregation mechanism in which the subscriber nodes have the ability to aggregate messages along the route to the destination. This proposal requires a prior knowledge of the network topology or using a service discovery mechanism. This becomes one of the disadvantages of this proposal since the topology of a WSN is generally dynamic. In MIRES [55], data aggregation is performed in publisher nodes. These nodes conduct in-network data reduction which results in reducing the number of message transmissions, latency and power consumption.

2.5 Timeliness

Protocols that provide this feature make use of the priority. It consists of making use of queue management to send the packet with the highest timeliness requirements to the

destination first. [68,69,76,77] are examples of these cases. [66] provides timeliness through a deadline mechanism. It allows discarding packets before reaching destination if their deadline has expired. These proposals do not also consider network parameters such as RTT that allow knowing the conditions of the network at a given time. Sending packets to the destination in conditions where the network is congested initially causes a greater delay that will result in a late packet delivery at the destination. This results in wasting of resources such as energy and bandwidth, which are very restricted in this kind of networks.

2.6 Quality of Service

The reliability of packet delivery, data aggregation and timeliness above mentioned are some of the most important features of Quality of Service need on WSN in order to meet the requirements of applications. The provision of these features is going to depend on the type of messages the application has to receive. Therefore, it is important the design of mechanism or protocols that provide to the application a way to negotiate or express the features of Quality of Service required. Several approaches in WSN have been proposed in the literature. Most of them satisfy QoS reliability requirement, and others satisfy besides the timeliness requirements**.** In [78,79], authors propose QoS in two domains: reliability and timeliness focusing on network protocol and MAC layer. For reliability domain, they consider the sensor node density in the network to route packets for multiple paths using a geographic location protocol or GPS in the node. Therefore, they can guarantee with certain probability the required end-to-end reliability level. For timeliness domain, they allow to choose the proper speed options for packets in order to satisfy different end-to-end deadlines. As each node maintains a delay estimation to each neighbor, it can decide to guarantee different deadline targets. Nevertheless, an important disadvantage is the power overhead due to the increment of the amount of data transmission by transmitting duplicated packets. This approach could severely affect applications requiring a long network lifetime such as those based on Publish/Subscribe Model. In [80], authors propose a framework combining network and MAC layer called SchedEx-GA. In this case, the proposal guarantees the identification of a configuration to accomplish the QoS requirements. One of the drawbacks of this solution is the placement of more sink nodes to achieve the QoS required, when we cannot achieve a configuration. This solution is not compatible with publish/subscribe model for WSN. In [81] authors present a survey of several routing protocols providing QoS requirements. Most of them are considered for Real-Time Multimedia network so none of them considers the publish/subscribe architecture we are studying in our proposal. Our proposal focusses on Publish/Subscribe protocols that work on application layer to satisfy application QoS requirements. These research works mainly present mechanisms based on network and medium access layers, as well as cross-layer approaches. We consider that an approach at the application layer protocol is necessary because it provides to the application a way to express directly QoS requirements. A policy-based publish/subscribe middleware for sense and react applications is proposed in [12]. This middleware defines the delivery protocols of the packets generated by the publishers and subscribers with certain delivery guarantees: that is unreliable or reliable delivery. In PS-QUASAR [66], there is a proposal of a Publish/Subscribe middleware that implements timeliness, priority, and reliability as parameters for QoS support. The reliability is achieved through a retransmission mechanism, and the timeliness is accomplished by using a mechanism that implements deadline concept to allow discarding packets before reaching destination if their deadline has expired. MQTT-SN [22] subscribers are able to define three reliability QoS levels. First, a best effort delivery

service; second, with retransmission of a notification until receiver acknowledges it (duplicated packets can be received); third, a service that ensures that a packet is not duplicated.

CoAP [20] is a standard application protocol suitable for WSN published on RFC 7252. This protocol takes into account most of the problematics about the way to deliver packets in a reliable way in WSN. This protocol provides reliable packet delivery through two levels. In the first level, the packets are called NON packets, these ones are sent in best-effort mode. The second level uses CON packets, which require confirmation packets from the destination. The extension of this protocol on RFC 7641 [82] proposes that the publisher (called server) is the only one to decide the level of reliability to send the packets. There is no support for the subscriber (called observer) to negotiate or express the required level of reliability.

As it can be noted in Table 1 which shows a summary of reliability, data aggregation and timeliness from all revised proposal in this chapter. From this table we can conclude that most of the presented protocols provide reliable delivery packet using fixed RTO which could result in a unsuitable reaction or response to a loss of data packet. In addition, the priority queuing is the mechanism mostly used by the proposals, however this does not consider the network conditions such as experimented delay which would affect that the message meets the timeliness requirement of application. Finally, no one of them provides data aggregation mechanism which is an important technique to save critical resources such as energy and bandwidth on WSN.

Proposal	**Reliable Delivery**	**Data Agg.**	**Timeliness**
Sharifi, M. et alt [76,77]	X	X	Priority Queuing
Felemban E. et alt [78,79]	Fixed Retransmission		Priority Queuing
Mires [55]	X	X	X
Russello, G., et alt [12]	Fixed Retransmission	X	X
PS-QUASAR [66]	Fixed Retransmission	X	Deadline Mechanism
Hui-Ling, C. et alt [67]	Fixed Retransmission	X	X
TinyDDS [68]	Fixed Retransmission	X	Priority Queuing
sDDS [69]	Fixed Retransmission	X	Priority Queuing
SchedEx-GA	Fixed Retransmission	X	Priority Queuing
PSWSN-MM [81]	X	X	X
MQTT-SN [84]	Fixed Retransmission	X	X
CoAP [20]	Fixed Retransmission	X	X
CoAP+CoCoA [23]	Dynamic Retransmission	X	X
Our Proposal	Dynamic Retransmission	Lossless	Deadline Mechanism

Table 1. Publish/Subscribe protocols QoS features summary

In this thesis, we propose a complete functionality of these features and take advantage of data aggregation to be efficient with the WSN traffic. We provide three QoS levels to meet application requirements in WSN. In the first one, we improve our first mechanism to compute RTO in dynamic way to provide packet delivery reliability. The second one adds data aggregation to reduce collisions and energy consumption and the last one provides timeliness using a deadline mechanism

2.7 Protocols

In this section, we examine the main features and functionalities of the protocols on which is based the research carried out in this thesis.

2.7.1 IEEE 802.15.4

This standard is intended for low LR-WPAN networks (Low-Rate Wireless Personal Area Network) where nodes operate with limited battery and data rates. It defines the physical layer (PHY) and medium access control (MAC) sublayer specifications for this kind of network [31].

Following are the most relevant capabilities provided by the standard relevant for this thesis:

- Star or peer-to-peer operation
- Unique 64-bit extended address or allocated 16-bit short address
- Optional allocation of guaranteed time slots
- Data rates of 250 kb/s, 100 kb/s, 40 kb/s and 20 kb/s
- 16 channels in the 2450 MHz band, 30 channel in the 915 MHz band and 3 in the 868 MHz band.
- Carrier sense multiple access with collision avoidance (CSMA-CA)
- Link quality indication
- Fully acknowledged protocol for transfer reliability
- Low power consumption
- Energy detection

There are two types of devices defined in these networks. The first ones, are called Full Function Device (FFD). This device can be used as PAN coordinator, coordinator or as device. The other kind of devices are known as Reduced Full Function (RFD). In this case, this device is intended for applications which do not need to send large amount of data. In addition, these devices can only associate with a single FFD at a time and can be implemented using minimal resources and memory capacity.

Figure 11 shows the two topologies for network connection supported by this standard: peer-to-peer topology and star topology.In peer-to-peer topology any device is capable to communicate with any other device as long as they are in range of one another. In this topology, message are routed through multiple hops from any device to any other device on the network. In addition, it allows more complex network formation such as mesh network topology and also complex applications such as: industrial automation, assets tracking and intelligent security surveilance.

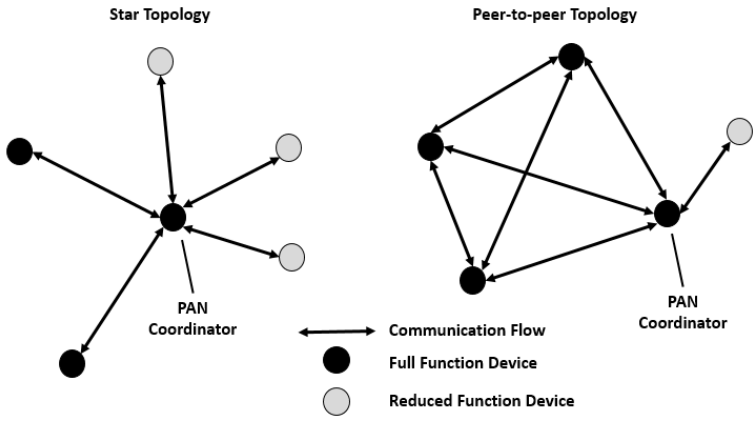


Figure 11 Network connection topologies for LR-PAN networks

On the other hand, star topology requires a single central controller known as: PAN coordinator. This device is in charge of initiating, terminating, or routing communication around the network. Generally, devices on this network topology will be battery powered, however the PAN coordinator will be mains powered.

Regardless of the network topology used, the devices have a single address known as extended addresses with a size of 64 bits. However, each PAN has associated with a unique identifier (PAN-ID) that allows devices to communicate within its PAN using the short address format formed by 16 bits.

The networks based on this protocol have the ability of allowing devices to detect their surroundings, cooperate to form topologies, and monitor and adapt to environmental changes, without human intervention which is refered as self-organizing feature. Also, this kind of network are self-healing, which refers to the capability of discovering, diagnose, and react to network disruptions and start corrective actions to recover the network or a node.

In the latter mechanism called slotted CSMA-CA when a device wishes to transmit should align it with the start of the beacon transmission of the PAN coordinator. Our work in this thesis is only based on non-beacon enabled PAN, therefore this mechanism is out of the scope of this thesis.

2.7.2 MQTT-S (SN)

MQTT-SN previously called MQTT-S is an extension of the publish/subscribe protocol Message Queuing Telemetry Transport (MQTT) and is optimized for the implementation on networks with limited bandwidth of 250 kbps, processing of a very short length message of 127 bytes, storage resources such as wireless sensor networks.

MQTT-SN is agnostic of underlying networks services, MAC and physical layer provided by IEEE 802.15.4 protocol. Furthermore, MQTT-SN can operate on any network technology that provides a datagram service. MQTT-SN does not require the connection-oriented transport provided by TCP (Transport Control Protocol), so it is well-suited for use over UDP. A MQTT-SN message consists of two parts: a 2 or 4 octet long header and an optional variable part. The size of the variable part depends on the type of message. MQTT-SN architecture is composed of three components as showed in Figure 12.

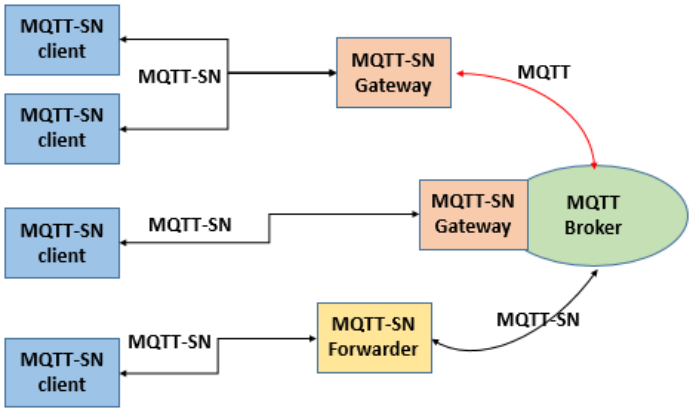


Figure 12 General MQTT-SN architecture

Firstly, a node called MQTT-SN gateway to connect with a MQTT broker in the external network. This node is located at the edge of the WSN and is in charge of message mapping between the MQTT (for external network) and MQTT-SN (for WSN) protocols. A MQTT-SN gateway may or may not be integrated within the broker. Secondly, nodes called clients which connect to MQTT Broker through the MQTT-SN Gateway. Finally, nodes could connect to MQTT-SN gateway through nodes called MQTT-SN forwarder. This component encapsulates the MQTT-SN frame it receives, and forwards it without changes to the MQTT-SN gateway. The process is similar in the opposite way.

In this architecture, broker node is responsible for managing subscriptions as well as storing and sending publications to corresponding WSN subscriber nodes. The client nodes are sensor nodes of WSN able to act as publisher, subscriber or relay nodes in case of a multi-hop scenario, in order to establish connection with the broker node.

As we noted in the Figure 12, the MQTT-SN architecture considers that the nodes (MQTT-clients) in the WSN will always act as sources of information (publishers) and that the interested parties will be in the external network, which results in the flow of traffic always increasing in the upward direction (from the WSN to the MQTT Broker Node). However, the situation is different under paradigms such as the IoT that we have already mentioned, where traffic can remain inside the WSN since an object (subscriber) may be interested in the information of another object (publisher). This thesis focuses on traffic inside WSN and therefore a general architecture from our point of view is showed in Figure 13.

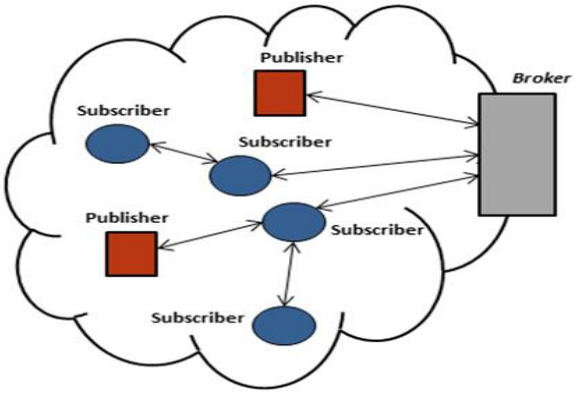


Figure 13 General MQTT-SN Architecture from our point of view

MQTT-SN provides a reliability mechanism consisting of three QoS levels. The first one, called QoS level 0, offers a best-effort delivery service, and no retransmission or acknowledgements are defined. The QoS level 1 is the second level which provides data recovery through retransmission of messages until they are acknowledged by the receivers. However, this QoS level does not prevent duplicate reception, which is referred as the messages arriving at the destination multiple times because of retransmissions. The last level is called QoS level 2. This level provides the same feature as QoS level 1, however it ensures also that the messages are delivered only once to the destination by using a four message handshake.

Since QoS Levels 1 and 2 require a retransmission mechanism, MQTT-SN defines two parameters in its best-practice guidelines [84]: a fixed RTO (between 10 and 15 seconds) and a maximum retransmission number (between 3 and 5 seconds). Since there is no reference to an application scenario in these best-practice guidelines, we use the lower and upper values of the RTO for evaluation purposes.

In addition to reliability mechanism, MQTT-SN uses a "stop and wait" mechanism for the transmissions of publication messages with QoS Levels 1 and 2. This means that any point in time a publisher node has to wait for the termination of its publication message flow with the broker node before it can start a new one. A publication message flow finishes when the publisher node receives the corresponding confirmation message according to the QoS level. This situation is repeated between the broker node and the subscriber node.

According to this way of working, a publication discipline for QoS Levels 1 and 2 should be provided to handle the publication messages generated while a publish message flow is in progress. It should be pointed out that the RTO is active while the publication message flow is in progress. The MQTT-SN specification [15] does not indicate the way to handle this situation. One way to address this issue would be for the publisher node to queue the new message publication until the confirmation message is received from the current publication message flow. However, due to the limited memory resources of sensor nodes, this aspect becomes a critical operational factor. Accordingly, in the experiments carried out to evaluate this publication discipline we do not consider the queuing mechanism. Instead, if there is a pending confirmation of a previous publication message, the new publication message is discarded. We refer to this discipline as the "persistent mode" discipline, because it always attempts to retransmit the earlier publication message and receive the confirmation message instead of sending the new one.

2.7.3 CoAP

CoAP[20] is a RESTful (Representational State Transfer) protocol for interaction with constrained devices and networks such as WSNs. This protocol was developed by IETF and is based on REST architecture which allows accesing to Internet resources by an application process and identified by Universal Resource Identifiers (URI). It is built on top of the UDP and therefore has a significantly lower overhead than TCP (Transmission Control Protocol) as shown in Figure 14.

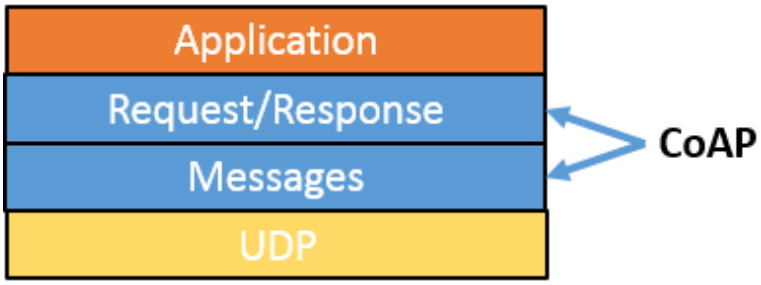


Figure 14. CoAP Protocol Stack

In addition, CoAP defines two kinds of interactions between end-points: 1) The client/server interaction model, where request messages initiate a transaction with a server, which may send a response to the client with a matching transaction ID, and 2) A publish/subscribe interaction model called the observer model [19], where a server (publisher node) can send notify messages (publications) to an observer (subscriber node) about a resource (event) that the subscriber is interested in receiving.

The client/server interaction model is based on the polling method. As mentioned earlier, this method is unsuitable for applications requiring information in real-time in order to react when an event of interest occurs. Therefore, we focus only on the publish/subscribe interaction model provided by CoAP, which is illustrated in Figure 15. With this model, CoAP allows a subscriber constantly to observe the events. This is done by the subscriber registering its interest in the event by means of an extended GET request sent to the publisher node. The publisher node establishes an observation relationship between the subscriber and the event, i.e., subscription is performed. The publisher notifies each subscriber node that has an observation relationship with the event.

Although in the general architecture of this model the publisher node also plays the role of broker, CoAP also enables high scalability and efficiency through a more complex architecture, which in fact supports the use of caches and intermediaries (proxy) nodes that multiplex the interest of multiple subscribers in the same event into a single association, as shown in Figure 15.

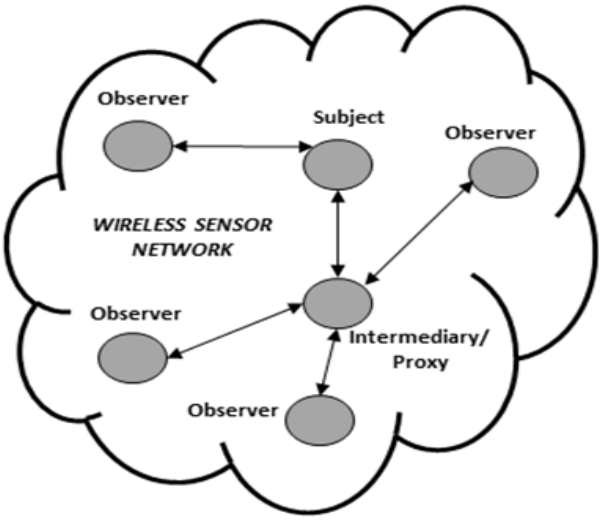


Figure 15. Architecture of the CoAP Observer Model

Besides, to overcome limitations in constrained devices and networks, CoAP provides optimization in the length of datagram and provide reliable communication. CoAP defines a retransmission mechanism to compensate for the unreliability of UDP protocol through two types of messages: 1) A Confirmable (CON) message: the message is retransmitted if no delivery acknowledgement was received, 2) A Non-Confirmable (NON) message: in this case there is no need to acknowledge the message. Besides, there are other messages such as: RST (Reset) which indicates that the server is not able to process de CON message and finally, ACK (Acknowledgment) message which is sent by the server as a reply of a CON message.

In the case of CON messages, the reliability mechanism uses an RTO fixed mechanism. This consists of setting an initial RTO value to a random number between an ACK TIMEOUT constant and an ACK TIMEOUT multiplied by ACK_RANDOM_FACTOR constant [20]. The messages that have not been acknowledged within RTO duration are retransmitted and subsequently this RTO value is doubled (exponential back-off mechanism). CoAP also defines a MAX_RETRANSMIT constant, which specifies the maximum number of message retransmissions. Table 2 shows the corresponding values for each protocol constants that are defined in [12].

Protocol Constants	Value
ACK_TIMEOUT	2 seconds
MAX_RETRANSMIT	4
ACK_RANDOM_FACTOR	1.5

Table 2. CoAP Protocol Constants for message transmission

Compared with MQTT-SN, we are able to state that Non-confirmable CoAP messages correspond to messages of QoS Level 0 in MQTT-SN and Confirmable messages are similar to QoS Level 1. Since the CoAP protocol has only two types of QoS, the QoS Level 2 of MQTT-SN is not evaluated in this article.

CoAP also uses a “stop and wait” mechanism for the transmission of CON messages. Therefore, in a similar manner to MQTT-SN, this protocol also requires a publication discipline to handle publication messages generated while the publication message flow is in progress (RTO is active). In this context, the publication discipline of CoAP will be activated when the publisher node wants to notify the subscriber node of a change in the state of the event. Thus, it must stop the retransmission of previous publication message and transmit the new one with the number of attempts remaining from the previous publication message.

Finally, CoAP has proposed a new mechanism called CoCoA (CoAP Simple Congestion Control/Advanced) [23] that potentially improves congestion control mechanism. This mechanism works with two estimators in parallel to calculate the RTO called RTO strong and RTO weak. The first one is calculated when an ACK is received after the first transmission of a packet. RTO weak is computed when an ACK is received after the first retransmission of a packet.

Finally, CoAP messages are composed by header with fixed length of four bytes, plus a variable-length Token, followed by options with a variable length and a payload prefixed by a payload marker which indicates the end of the options and the start of the payload, as is shown in Figure 16.

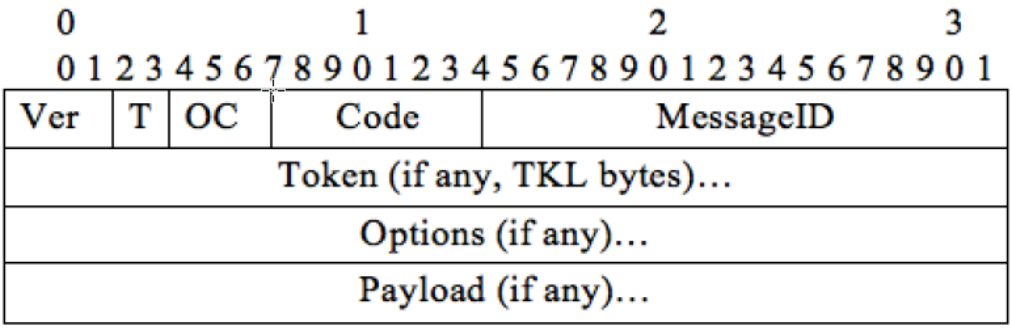


Figure 16 CoAP Message Format

3. Presence Service and Publish/Subscribe Architecture for Wireless Sensor Networks

In this chapter, we first present our results about the evaluation of presence service protocols on Internet such as SIP/SIMPLE and XMPP, mainly focusing our attention to the fact that they should be used over wireless environments and resources-constrained devices such as sensor nodes. As we mentioned, these protocols are not suitable to be implemented directly in the WSN, because these networks have limitations in terms of processing capacity, bandwidth and energy. This results in the need for a protocol that makes efficient use of these resources to prolong the life of the network and to optimize the network communication. Considering the results of this evaluation, we propose several design points to consider for a presence service protocol on WSN.

We continue in this chapter with the analysis and evaluation of existing publish/subscribe protocols on WSN because presence service technologies are based on this model. We focus on important design aspects that impact the resources of WSN such as: reliability, sleeping support and messaging format to save energy. This evaluation results in our proposal for a presence service architecture for WSN based on publish/subscribe protocol. Our proposed architecture provides mechanism addressed to save energy in the network communications with the use of data aggregation and publication on demand techniques. Also, it provides more scalable and flexible network architecture because the network could be formed by broker domains.

We finalized this chapter with a proposal of an assisted living system for home called Presence-Aware Smart Home (PASH). This system consists of several components with the role of broker, publisher and subscriber. We apply all of mechanism we proposed in our previous presence-based architecture for WSN based on publish/subscribe paradigm.

We demonstrated that the sharing of presence information coming from sensors inside a home such as temperature, gas sensor, light sensor and others could detect potential risky situations, and actuator nodes could perform actions for mitigating the risks.

3.1 Evaluation of Presence Service Technologies for WSN

In this section we present the obtained results of evaluation of the most extended standard existing on Internet such as SIP/SIMPLE and XMPP, considering their ability to adapt to wireless environments, the size of its messages and its energy efficiency, all of them requirements in WSN.

With reference to wireless environment adaptation, SIP/SIMPLE can use TCP (Transport Control Protocol) or UDP (User Datagram protocol) as transport protocol. The ability to use UDP is a great benefit on such environments because is lighter and faster than TCP because UDP does not check for errors and does not guarantee delivery. However, UDP is not the solution for every situation; we must consider the requirements of applications. For instance, in some critical reliability scenarios it would be preferable to guarantee delivery of a message than to increase the speed of message transmission.

Unlike SIP/SIMPLE, XMPP works only over TCP. Therefore, the considerable header overhead, connection management, end-to-end flow and congestion control lead to TCP poor performance on wireless environment and resource constrained nodes, such as WSN.

Regarding size of message used for both protocols, SIP/SIMPLE is based on request/response model with text messages. Each message contains ASCII strings and allocates one byte for each character and that results in long messages to fit into the size of the message of sensor nodes.

On the other hand, XMPP is a protocol based on XML (eXtensible Markup Language) that makes it quite chatty to transmit and requires some computation capacity to parse and interpret messages than could be complex in resources-constrained devices, such as sensor nodes.

Regarding energy efficiency, transmitting and receiving data consumes energy proportionally to the size of the message. Hence the large size of the messages both SIP/SIMPLE and XMPP are not suitable for sensor nodes generally working with a limited energy resources (batteries).

On the other hand, in general, presence service on Internet has certain limitations to be used on WSN such as:

- Limited vocabulary to express presence information: A simple status of presence information (such as: offline, busy, away) is not enough to decide how to establish a communication with an entity. Besides regarding interaction with objects there is a need to enrich vocabulary. This could be accomplished using context information that sensor nodes can collect.

- The presence status is generally updated manually: there is not a way to deduce automatically presence status of a user. Some presence technologies can change presence information status to "away" taking in an account there is not activity from the user in the device during some period.

- In the case of sensor nodes, should be necessary that they change their status without user interaction.

Considering the issues evaluated in this section, the results indicate that existing protocols for presence service could not be implemented directly in the WSN because they do not consider aspects which are specific characteristics of this type of networks, such as: energy efficiency, and optimization of communication. However, both standards protocols (SIP/SIMPLE and XMPP) use the publish/subscribe communication model, which also use certain protocols in WSN not intended for presence services on WSN.

3.2 Evaluation of Publish/Subscribe protocol in WSN for Presence Service

In this section, we make a technical comparison among protocols already known in sensor networks, with a focus on the most important design points that impact the resources of WSN.

A. Topic composition

MQTT-SN allows constructing different types from topics. In addition, a topic can include information of several nodes by means of the use of "wildcard". For example, if we have interest in obtaining data of all the sensors of temperature located at the first floor of a building we should subscribe us to the following topic: "sensors/floor1/+/temperature". This form to operate allows that only it must transmit a message SUBSCRIBE, resulting in an energy saving for the sensors nodes.

If we apply this situation to TinySIP, it will require to send a SUBSCRIBE message for each node temperature sensor located in the location of interest result in increasing of message traffic.

B. Sleeping nodes support

Since the WSN nodes usually have a limited power source, these devices need to enter a sleep mode, and will wake up whenever they have data to send or to receive. In this sense, MQTT-SN incorporates a mechanism by which clients indicate the time they will be inactive at Gateway/broker, and therefore it will buffer messages destined to them for later delivery when they wake up. In contrast, TinySIP does not consider this particularity from the nodes from WSN, resulting in lost messages that are sent to the nodes that are inactive.

C. Reliability

TinySIP does not specify any reliability mechanism. However, MQTT-SN defines three QoS levels. QoS level 0 is the simplest one: it offers a best effort delivery service, in which messages are delivered either once or not at all to their destination. No retransmission or acknowledgment is defined. QoS level 1 provides a more reliable transport: messages are retransmitted until they are acknowledged by the receivers; however certain messages may arrive multiple times at the destination because of the retransmissions. The highest QoS level, QoS level 2, ensures not only the reception of the messages, but also that they are delivered only once to the receiving entities.

D. Architecture

Both, TINYSIP and MQTT-SN need to incorporate a called element Gateway that makes the function of translating the messages that come from the WSN towards the external network or vice versa, so that they are compatible between his respective protocols and extensions (SIP to TINYSIP and MQTT-SN to MQTT and vice versa).

MQTT-SN additionally needs a component called Broker that is the one that manages the connections and the subscriptions of the clients (sensors nodes) and it also notifies the application when a publication exists on an interest topic. Nevertheless, the Broker can be integrated along with the Gateway in a same location.

Also, both protocols allow the use of multiples Gateways, with certain differences. MQTT-SN allows that a sensor node can be connected to another Gateway in case the sensor node has lost communication with a previous Gateway. In the case of TinySIP, a sensor node could be contacted simultaneously by anyone of the Gateways in which this node is registered.

E. Messaging Format

Both protocols use a reduced size messages to be compliant with constrained-resources nodes on WSN. For MQTT-SN, it considers that messages should be no longer than 64 bytes payload of an 802.15.4 message. However, TinySIP uses a more compact message format so that a TinySIP message can be transmitted within as smaller payload, such as the 29-byte of a regular TinyOS payload message.

We have analysed the amount of messages that must be exchanged in each phase by both protocols, we can note the following:

The first phase consists of setup connection. With respect to MQTT-SN, it requires a client to setup a connection with the Broker before it can exchange publications and subscriptions with the Broker. To this end, client transmit firstly a CONNECT message including "client_id", and parameter connections. Besides, it transmits optionally two additional messages: one to indicate the "Will_Topic", and another message to indicate the "Will_Message".

Both messages are transmitted from Broker to subscribers when it abnormally loses the connection with the client (sensor node). For each message there is a confirmation one from the Broker to the client. Therefore, at the worst case, it will exchange six messages between each client and Broker, as shown in Figure 17.

On the other hand, TinySIP does not include this phase, resulting in a less amount of messages for its communication.

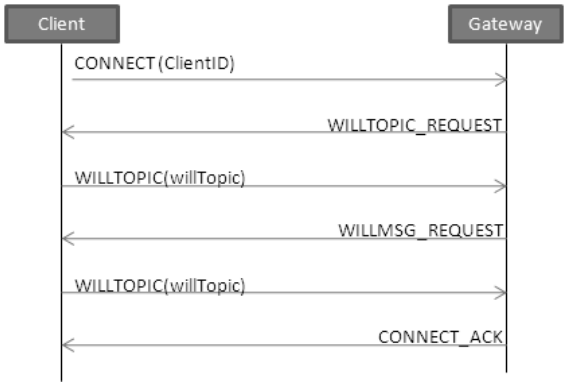


Figure 17 Setup connection for MQTT-SN

The second phase consists of a register topic, as illustrated in Figure 18. Both protocols require that each client indicates to Gateway or Broker the events it can sense. For MQTT-SN, once client informs its topic to Broker, the Broker assigns to this a topic_id of two-byte long. However, we can predefine topics, so that this phase can be omitted. On the other hand, TinySIP uses a mapping between traditional REGISTER SIP message and a coded message TINYSIP-REGISTER to fit into 29 bytes message size.

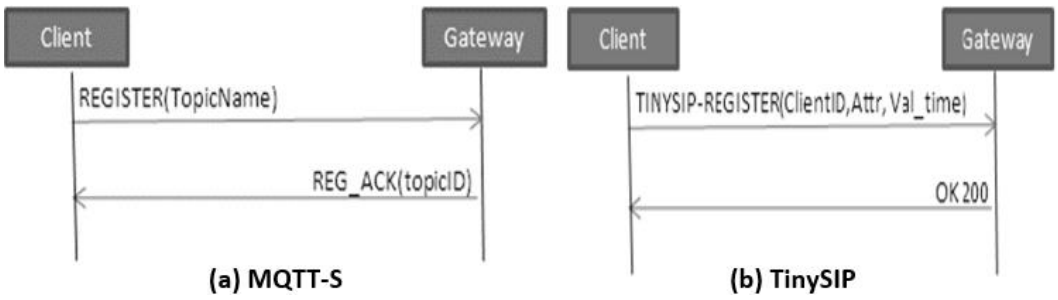


Figure 18. Register Topic procedure

The third and fourth phases consist of publishing and subscribing topics. Each client (sensor node) publishes data corresponding to topic if previously any application has

subscribed to this. Table 3 summarizes the number of messages implicated in each phase for both protocols.

Phase	MQTT-SN	TinySIP
Setting up Connection	6 messages	N/A
Registering Topic	2 messages (optional)	2 messages
Subscribing Topic	2 messages	2 messages
Publishing Topic	2 messages	2 messages

Table 3. Amount of messages exchanging for each protocol

3.3 Presence-based Architecture for Wireless Sensor Networks using Publish/Subscribe paradigm

In this section we present our proposal for a presence-based architecture considering the relevant features of current publish/subscribe protocols and the requirements above mentioned.

Unlike traditional centralized architecture where there is only one broker node located between WSN and the external network making the sink node functionality, our proposal is focused on distributed architecture where there are several broker nodes to reduce the bottlenecks that lead to low network performance.

In addition, centralized architecture is unsuitable for providing a presence service on WSN because these protocols consider the communication will only occur between sensor nodes and external users or vice versa. Thus, all the information is sent to the broker node (sink) located between WSN and external network. In contrast, our proposal is focus on trends such as Internet of Things and presence service in WSN where objects and devices can interact among them and not only with the others located on external networks.

Our architecture consists of three components of Publish/Subscribe Paradigms [85] which are: publisher nodes, subscriber nodes and broker nodes. Broker nodes are distributed inside WSN aside from the broker node acting as sink node. Figure 19 depicts this distributed architecture where there are, as example, two broker nodes inside WSN, and another broker node at the edge of the network.

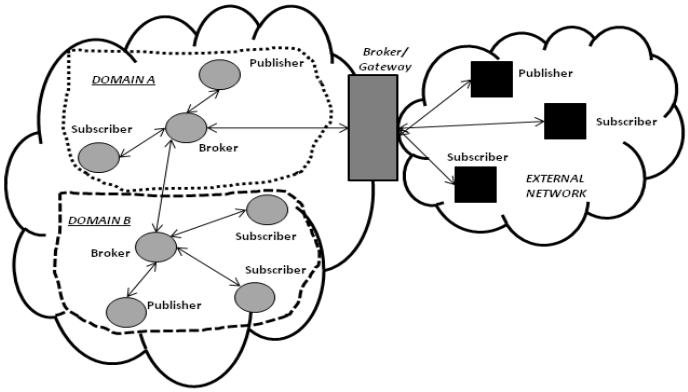


Figure 19 Proposed Presence Service Architecture for WSN

This one is also acting as gateway to communicate with the external network. Thus, it allows distributing the load in the network by grouping publisher and subscriber nodes in different broker nodes.

When a potential publisher or subscriber node enters into the network, it sends a broadcast message to discovery the nearest broker node. Once the publisher or subscriber node receives a reply, it will establish a connection with the broker node. The group of subscriber and publisher nodes managed by a broker node we call broker domain. The broker nodes will be the only responsible to exchange message between broker domains, as we can see in Figure 20.

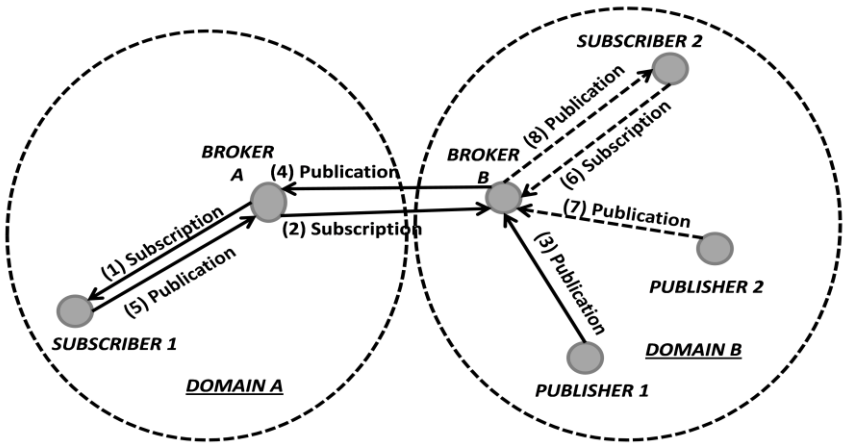


Figure 20. Broker roles and broker domain

Also, a broker node might play a publisher or subscriber role on behalf of its domain. Figure 20 shows this situation. “Subscriber 1” and “Publisher 1” are placed in different broker domains. “Subscriber 1” on broker domain A wants to receive presence information from “Publisher 1” on broker domain B. So, “Subscriber 1” makes a subscription (1) in its respective broker node (“Broker A”). This one makes a subscription (2) in the broker node responsible for broker domain B (“Broker B”). When “Publisher 1” sends presence information (3) to its broker node, this one will forward it (4) to broker node of Broker Domain A, and this one; will transmit it (5) to the “Subscriber 1”. In this way, nodes located in different Broker’s domains will exchange messages through corresponding broker nodes. It should be noticed that a broker node learns the identities of brokers in other domains. That is because it receives a broadcast message of available topics to subscribe in each broker. Then, it will store this information to be available when a potential subscriber node wants to subscribe to a topic. On the other hand, publisher and subscriber nodes located in the same broker domain, will exchange messages through the broker node responsible for this domain. Thus, exchanged traffic is kept inside the broker domain. This situation is also depicted on Figure 20, where “Subscriber 2” makes a subscription (6) through “Broker B” to receive presence information from “publisher 2” on the same broker domain B. When “Publisher 2” publishes presence information (7) to its broker node, “Broker B” will forward it (8) to “Subscriber 2”. In conclusion, the proposed architecture contributes to isolate the exchanged traffic in broker domains and will exchange traffic between broker domains through corresponding broker nodes only when it is needed.

3.3.1 Data Aggregation

In our architecture, we make use of wildcards in similar way as MQTT-SN [22] does it. This allows that we can transmit only one SUBSCRIBE message to subscribe us to a topic related to several publishers. In addition, we propose two ways to perform data aggregation with the purpose to decrease the number of messages into the network. Data aggregation will be performed either in the publisher or in the broker node.

Data aggregation in broker node: The broker node aggregates publication of concerning publishers (i.e. by configurable function: average, min, max, status, etc.). Then, it transmits only one packet to the subscriber nodes. Data aggregation in publisher node: we propose the publisher nodes can aggregate packets in a "lossless way". That means that publisher will not transmit publications to the broker node until a timer expires. They will aggregate in a single message all publication events produced during the timer period. Then, they will create only one packet to transmit to the broker node. This technique results in reducing the number of transmissions and thus energy saving of network nodes. However, there is a trade-off between the number of transmissions and the delivery delay. The more the number of gathered data; the less is the number of transmissions. Nevertheless, waiting for more data increases the data delivery delay. This issue is addressed by the new proposed architecture. Therefore, we recommend this technique only be applied in situations where network congestion is detected or an application parameter has been defined. We can see the algorithm in Figure 21, where a publisher node detects a congestion indication (i.e. received from the broker or from MAC layer) or there is an application parameter defined. Then, the publisher node will activate a timer, we call "aggregation timer". During this period, publication events produced locally will be buffered and marked for "lossless" aggregation.

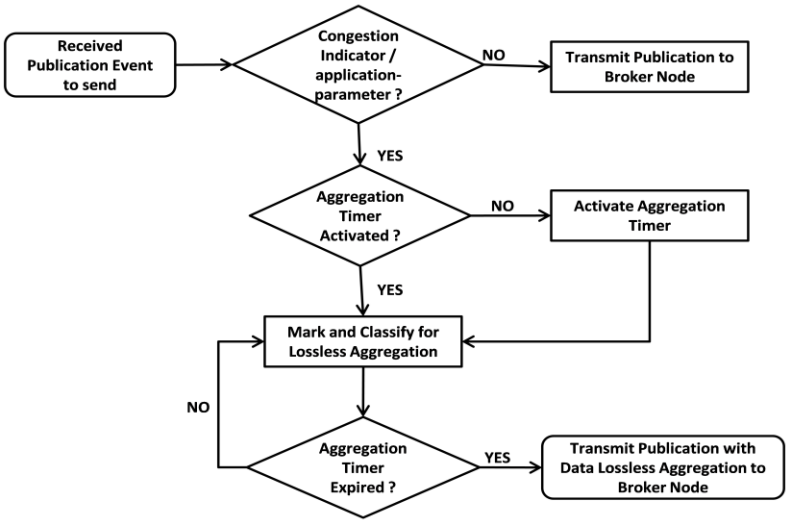


Figure 21. Lossless Aggregation Algorithm

The number of packets to be buffered will depend on memory resources. Also, the amount of data that one packet can carry will depend on the packet size limit. For example, 127 bytes is the maximum size of PHY service data unit defined in IEEE 802.15.4 [9] standard. When

the “aggregation timer” expires, the publisher node performs the lossless aggregation, and transmits the packet to the broker node. In case of congestion network, this mechanism prevents the publisher nodes continue transmitting packets. Thus, it avoids the increment of network congestion and the extra waste of energy. That is because of, during congestion, the transmission of packet would result in packet loss, thus extra packet retransmission.

3.3.2 Publication on Demand of Presence Information

Generally, publish/subscribe protocols allow a publisher to publish data regardless of the presence of a subscriber. This would result in extra power consumption at publisher node. Moreover, unnecessary network traffic increases the risk of network congestion. To overcome this situation, in our architecture, publisher nodes only send publications when there is a subscriber. So, the broker sends a notification message to the publisher the first time it receives a subscription. That is shown in Figure 22.

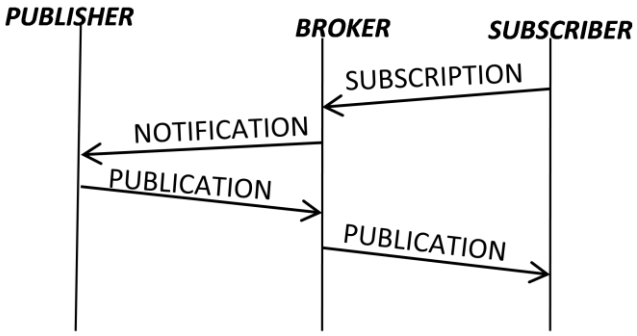


Figure 22. Publication on demand

Also, similar notification message is sent to publisher when the last subscriber decides to unsubscribe to presence information event (un-subscription). In this case, the publisher node stops events of presence information publication. This mechanism is referred as publication on demand of presence information. It should be noted that previously publishers communicate to the broker the events they are able to publish. This way, a broker knows about the publishers to send them the notification of subscription.

3.3.3 Advertisement of presence information to subscribers

Another important issue in presence services over WSN is that subscriber nodes receive presence information as soon as this has been published. The proposed architecture allows publisher nodes to include presence information (initial value) when they register the topics at the broker node as is showed in Figure 23(a). This presence information will be stored in the broker node and will be available immediately in the subscriber.

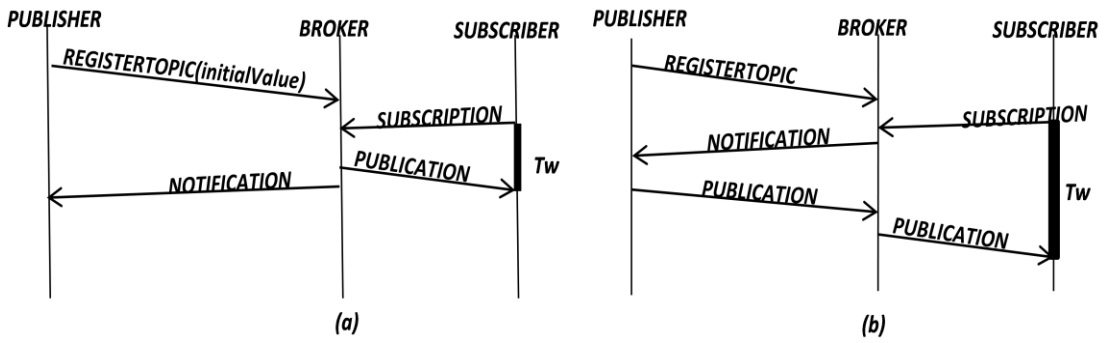


Figure 23. Spent Time in the first publication

Let suppose node 1 is the first subscriber for the registered topic A. Now, let consider Tw as the waiting time experimented by subscriber node 1 to receive the first publication. Tpb is the time used by the broker node to process the subscription of topic A and transmit the publication message to subscriber node 1. Then,

$$Tw = Tpb \qquad (3.1)$$

Otherwise, if publisher node does not register topic with initial value as it is showed in Fig. 3.7(b), Tw would be calculated by,

$$Tw = Tnp + Tep + Tpp + Tpb \qquad (3.1)$$

Where Tnp is the time used by the broker node to send a notification message indication to publisher to initiate the publication for topic A; Tep is the time until the next event for topic A is produced; and Tpp is the time used by publisher node to generate publication message and transmit it to the broker node.

3.3.4 A Presence-aware Smart Home System (PASH)

In this section, we propose an approach to a system that applies the mechanisms discussed above considering a general AAL scenario[86] Therefore, we describe the components and functionalities of the Presence-Aware Smart Home (PASH) system. This system aims to provide assistance to people in their home through "smart devices or objects" capable of sensing, measuring, communicating and generating some action to provide security and comfort in the home. For instance, Figure 24 shows an example of a scenario where the user is supported to live in a more confident, safe and secure way. In this case we suppose a smoke detector sensor has detected a fire event, so the actuator nodes switch off immediately the gas service and the electricity service. It shows how the home environment is able to detect potential risky situations and perform actions for mitigating the risks.

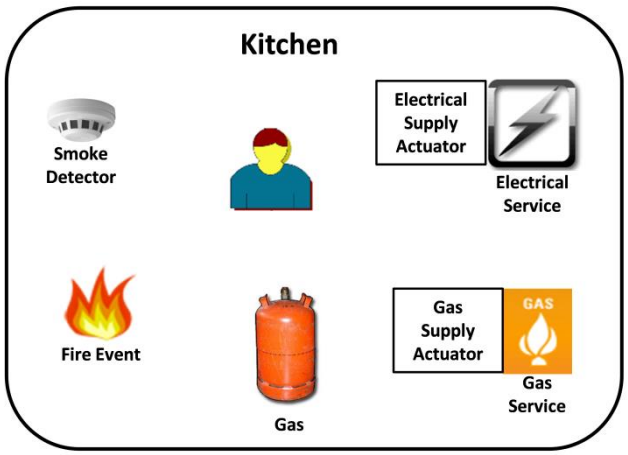


Figure 24. A general view scenario with smart devices in the kitchen.

The PASH system is composed of three components as shown in Figure 25. These components use the publish/subscribe communication model to exchange presence information.

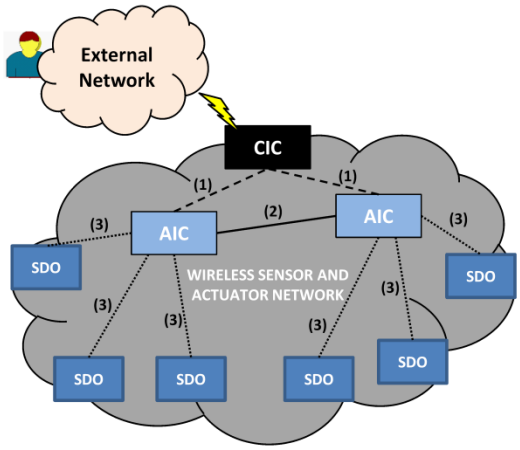


Figure 25. PASH System Components

1. **Smart devices or objects (SDO):** They are nodes used for detecting events, sensing the environment and performing action. They are commonly positioned in objects such as windows, doors, chairs, etc. In these cases, they are battery-powered node, so efficient energy use is an important issue to consider in these nodes. On the other hand, there are nodes embedded in electrical devices so they use the energy supplied by these devices. The SDOs only exchange messages with their corresponding AIC in the area. The circle dotted lines (3) in Figure 25, depicts this situation. SDOs are responsible of registering its capabilities in the AIC, publishing and receiving the interested presence information, and performing a corresponding action.

2. **Central Intelligent Coordinator (CIC)**: It is a central station node (such as PC station). It has at least two network interfaces to act as gateway between home area network and external network (Internet). The CIC only exchanges messages with

AIC. Figure 25 depicts this communication with dash lines (1). The CIC is responsible of storing the capabilities (temperature, light, window opener, etc.) of smart devices or objects (SDO) managed by each AIC, providing administration interface for configuring event/action rules. That means that through this interface the user can relate the event to be detected by sensor nodes and the action that will be performed by the actuator nodes. The CIC distributes this information to the smart devices through the respective AIC.

3. **Area Intelligent Coordinator (AIC):** It is a node installed per section, area or room in the home. It will be always active and connected to household power and used to manage the communication between nodes located in the room, with the CIC and in some cases with others AIC. Figure 25 also shows the communication between CIC with solid lines (2) and the dash lines depict the communication with SDOs. The AIC is in charge of locally storing the capabilities (temperature, light, window opener, etc.) provided by the SDOs. This information will be sent to the administration interface of the CIC for configuring the event/action rules as we mentioned previously in the CIC component description. The AIC will act as a broker for the smart devices or objects in its room. That means, it is responsible of storing the status (presence information) of the SDO's capabilities and dispatching it to interested parties (i.e., others SDOs) in its area or in some cases to CIC or others AIC. The AIC is also responsible for data aggregation and reports on demand any information requested for monitoring or administration tasks to the CIC.

The components of the PASH system publish and receive event information (presence information) of SDOs in order to perform an action for the safety, security and comfort of the people at home. In addition, PASH system is based on distributed publish-subscribe architecture where there are several AIC components located in the different areas, sections or room of the home.

As we mentioned in previous section, each AIC acts as a broker between the SDOs intended to communicate. It means that presence information is only transmitted to AIC responsible in the area. This area is we called broker domain in our previous explained architecture. Then, this one processes information and sends to the interested SDO located in the same area. We can see this situation on Figure 26. For instance, the lights are switched off when nobody is seated in the chair and movement is not detected in the living room. In this case the pressure sensors on the chair (SDO) and movement sensor (SDO), both publish presence information to respective AIC on the living room, and this one processes and transmits it to interested actuator node in the lamp to switch off the lights. This way, the AIC reduces dependencies between interested parties (SDOs), since an SDO interested on event (presence information) do not need to know who is the SDO publishing presence information or the amount of SDOs that publish this information.

On the other hand, the PASH system allows the communication between AICs, in order to exchange presence information between two areas. That means that AIC exchanges messages with other AIC in behalf of any SDO of its area. For instance, (Figure 26), when a visitor rings the doorbell his/her image is transmitted to the TV that person is watching in the living room. To achieve this objective, the doorbell (SDO) transmits its presence information ("on" or "activated") to its corresponding AIC, this one processes information and transmits it to the respective AIC where the interested SDO (TV) is located. These ways

of working allow that only the necessary traffic of data flows through the network, whereas the local traffic is isolated in each area managed by AIC. This is a relevant feature to improve system performance.

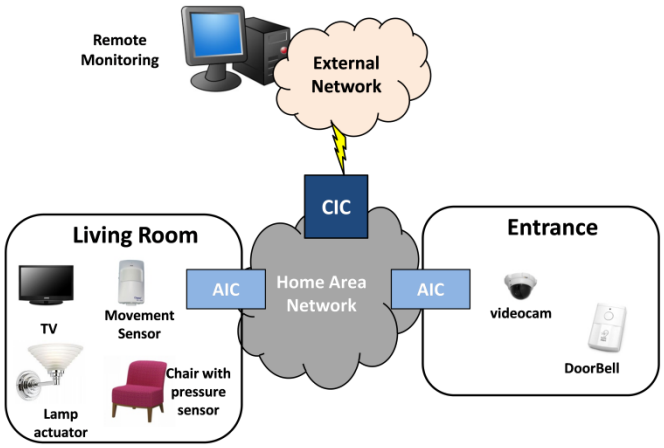


Figure 26. PASH's general architecture concept

PASH system is able to automatically detect new SDOs. That is, when a new SDO enters into the network broadcasts a discovery request message to find the AIC in the area. The AIC broadcasts a discovery reply message with information needed for SDO to establish a connection with the AIC, as shown in Figure 27. When connection is established between them, the SDO automatically registers its capabilities (temperature, light, window opener, etc.) to the AIC in the area, and this later one reports to the CIC for configuration through administration interface.

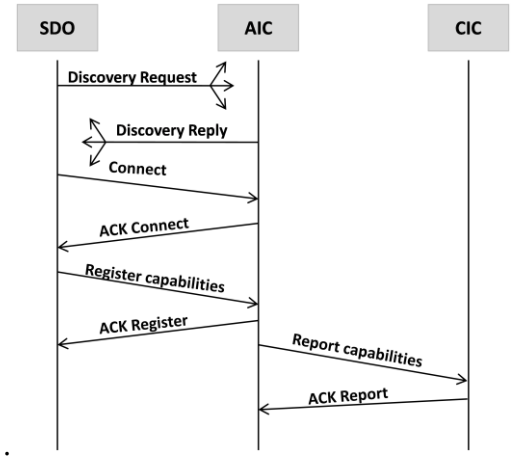


Figure 27. Example of AIC Discovery, SDO Registration and Report to CIC

PASH system addresses the energy efficiency of the nodes by performing two types of data aggregation depending on the type of component, either AIC or SDOs. In the AIC component, the event publication (presence information) received from SDOs will be

aggregated (e.g. by configurable function: average, min, max, status, etc.) and will be transmitted in case of data is aggregable. In this case, only one packet will be sent to the interested SDOs. This mechanism is related to broker node aggregation mechanism in our previous explained architecture. On the other hand, SDOs will not transmit event information to the AIC until a timer expires. They will aggregate in a single message all events publications produced during the timer period and will create only one packet to transmit to the AIC. This technique is based on mechanism detailed on algorithm depicted in Figure 21 of our explained architecture. As we mentioned, this technique results in reducing the number of transmissions and thus energy saving of the node. However, there is a trade-off between the number of transmissions and the delivery delay, since the more the number of gathered data; the less is the number of transmissions. This issue is addressed by PASH system; therefore, this technique only is applied in situations where network congestion is detected or when an application parameter has been defined by the user.

For monitoring purposes, the PASH system provides a mechanism for querying information in the whole network. This task is carried out through the CIC that is able to query each AIC about the presence information of any SDO in its area. All of this information can be accessed through the administration interface at the CIC. Therefore, this system provides to the user an easy central view of the SDO in the smart home. In addition, authorized users can access the administration interface from external network for remote monitoring, as shown in Figure 26.

3.3.5 Simulation Setup

We implement a simulation environment using the software OMNet++ [88] to make an evaluation of the proposed architecture. The network topology consisted of one broker node, one subscriber node and up to 32 publisher nodes depending on test scenarios. Each publisher node has a publish rate equal to 10 messages per second with a regular inter-packet interval. Each node is provided by an 802.15.4 network interface working in 2.4 GHz frequency. We focus our experiments in two proposed mechanisms: advertisement of presence information to subscribers and publication on Demand of Presence Information previously explained.

3.3.6 Results and Discussion

Firstly, Figure 28 depicts the results of comparing the spent time of subscriber node to receive publication. This aspect was used to evaluate the mechanism of advertisement of presence information to subscriber where publisher node is able to include initial value when it registers a topic. In this case, as we expected this mechanism reduces the waiting time around 38% comparing if we do not use it. This mechanism benefits to subscriber nodes because it does not have to wait the next time publisher node sends presence information.

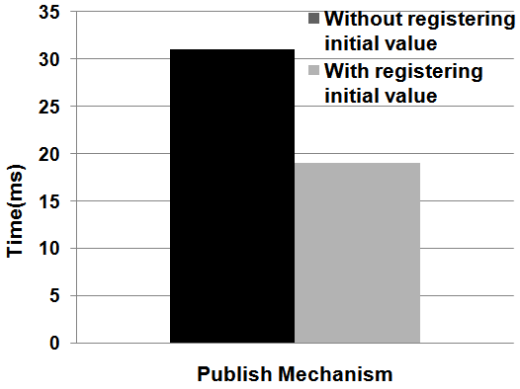


Figure 28. Spent Time until reception of the first publication by subscriber node

Next, we evaluated the publication on demand mechanism by examining the relation between the impact of the amount of publication messages without subscription in the network traffic when we increase the number of publisher nodes and the energy consumption when we increase the publication rate. From Figure 29 we can note a proportional direct relation between the number of publish messages transmitted without subscription and the number of publisher nodes.

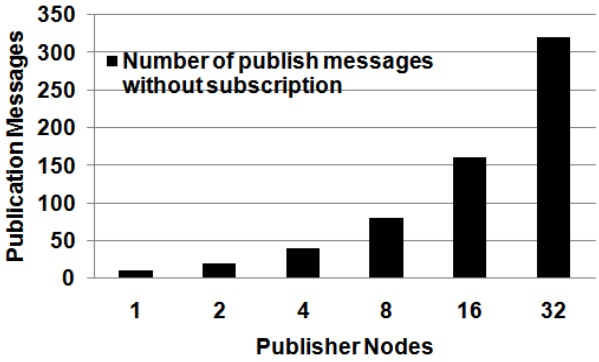


Figure 29. Publish messages transmitted without subscription

As we mentioned previously, this way to work has a high impact on energy resource of nodes and generates unnecessary traffic on the network. Therefore, the publication on demand mechanism has a great benefit. Figure 30 shows that our approach results in an energy saving between 3.6% for the lower publication rate and 80% for the higher one. There is an important benefit of using publication on demand mechanism.

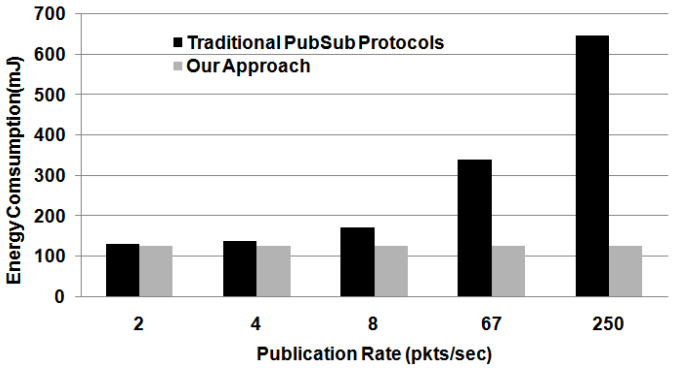


Figure 30. Energy Consumption of Publications on Demand

3.4 Conclusions and Contributions

Although SIP / SIMPLE and XMPP are the protocols mostly used to provide presence information in Internet, the results derived from the evaluation presented in this chapter indicate that these protocols could not be implemented directly in the WSN because they do not consider important aspects of this kind of networks such as: energy efficiency, and optimization of communication.

The use of publish/subscribe protocols in WSN is the alternative to provide presence service. In this chapter, we also evaluated the existing publish/subscribe protocols in WSN resulting in the lack of mechanisms such as reliability, optimization of communication and aggregation of data to reduce the consumption of bandwidth and energy. In addition, these protocols consider a communication between sensor nodes and external users or vice versa. However, trends such as "Internet of Things" consider objects can interact among them, thus publish/subscribe protocols should consider publisher nodes not always will send information to the sink as the only destination.

We have proposed a new architecture to provide presence service considering the most important aspects to deploy this service in WSN. In conclusion, the mechanisms provided by our architecture bring significant advantages with respect to other publish/subscribe protocols in WSN. The proposed architecture not only improves scalability but also provides mechanisms for an efficient energy management. The publication on demand mechanism prevents to transmit publication messages without related subscriber nodes. In addition, this architecture provides a mechanism to avoid that subscriber node waits for the next presence information produced by publisher node. It reduces waiting time of subscriber nodes. Results presented in this chapter show that the proposed architecture is suitable for WSN. The implemented "lossless" data aggregation algorithm would result in energy saving for publisher nodes. Moreover, it would reduce the traffic load in situations of network congestion and to long the lifetime of the WSN.

Finally, we have designed a system approach aimed to home control based on Ambient Assisting Living concept applying this architecture called PASH. The PASH system addresses the energy efficiency of WSN by providing several mechanisms to reduce the traffic in the entire network through data aggregation and publication on demand techniques. Unlike other projects, the PASH system distributes the intelligence among smart objects or devices and isolates the traffic by room or home area. Moreover, the discovery mechanism provides an easy way to configure and manage devices through devices. The

publish/subscribe communication model used by the PASH system allows scalability and flexibility, because of smart objects or devices publishing presence information do not need to know the details of the other devices interested in this information We also concluded that presence information provided by WSN nodes embedded on devices or objects can be used to perform action to facilitate the home living.

The results of the research presented in this chapter led to the presentation of a poster "*Presence Service for Wireless Sensor Networks: Research and Open Issues*" presented in the 9th conference of telematics engineering (JITEL) [P3], the conference article "*Presence-Based Architecture for Wireless Sensor Networks Using Publish/Subscribe Paradigm"*, that was presented and published in the 9th IFIP TC 6 International Conference on Wired/Wireless Internet Communications - WWIC 2011[P4] and the conference article "*A Presence-aware Smart Home System (PASH)"* that was presented and published in the III International Workshop on Ambient Assisted Living - IWAAL 2011 held at IWANN 2011[P5].

4. Reliability: Packet Delivery Evaluation on Publish/Subscribe Protocols

The fixed value for RTO calculation used by MQTT-SN and the CoAP (default RTO mechanism) protocols could be feasible for deployments where the RTT is close to the defined RTO value. However, a fixed RTO value is not compatible with the scalability and flexibility features provided by the Publish/Subscribe Model on WSN. A too short fixed RTO could lead to spurious retransmissions due to changes in network conditions, resulting in a waste of resources. Otherwise, it would cause the reliability mechanism to react too late to recover the packet loss. Considering the arguments above explained, in this chapter we present the discussion and results about a new mechanism with an adaptive RTO calculation method which could react properly to changing network conditions.

4.1 Simulation Environment

In this section we describe the simulation environment and performance metrics defined to evaluate both protocols.

Simulation experiments were carried out using OMNet++ [88]. We consider a potential application scenario of industrial automation. In this context, the goal of the application is the monitoring and control of critical parameters in a warehouse through a deployment of WSN.

We focus on one-hop scenario and multi-hop scenario that is generally in line with industrial monitoring and control [87] applications. Two types of devices are deployed in different parts of the application area: publisher and subscriber nodes. The publisher nodes are responsible for measuring the critical parameters in the warehouse. Additionally, there are two subscriber nodes, one of which receives publication messages in a best-effort mode for the monitoring process only. This means that the publication messages are received using the QoS Level 0 for MQTT-SN, and NON messages for CoAP. The other subscriber node receives publication messages in reliable mode for controlling the critical parameters. This means that it receives publication messages using QoS Level 1 (MQTT-SN) or CON (CoAP). Communication between these devices is in many cases carried out through a device acting as the central controller or gateway device. When the critical parameter is above a predefined threshold, this subscriber node acts appropriately. For instance, activating an alarm to evacuate the personnel or by turning on the ventilation system to avoid a risky situation. If in this kind of application, the reliability requirement is not met, the correct execution of control actions may be severely compromised.

For this critical application where the monitoring and early detection of critical condition is crucial, we consider that the data is generated periodically every second by publisher nodes and is then sent to the broker node. We also choose this data generation rate to study the system in stress condition. Since each subscriber node receives the publication messages with a different reliability level, the broker node has to receive the publication messages from publisher nodes with the maximum reliability level, as previously explained.

The number of publisher nodes varies from 10 to 100 in steps of 10. We use the term “broker” node to refer to the central node specified in the MQTT-SN architecture and the role of the proxy node for CoAP protocol.

To evaluate our proposal and compare with MQTT-SN and CoAP, we consider 3 network topology scenarios: single hop, extended single hop and multi-hop network topology. In the single hop scenario all nodes are within communication range of each other. Subscribers and publisher nodes are placed at the same distance from the broker to achieve fairness among nodes by preventing the capture effect, as illustrated in Figure 31.

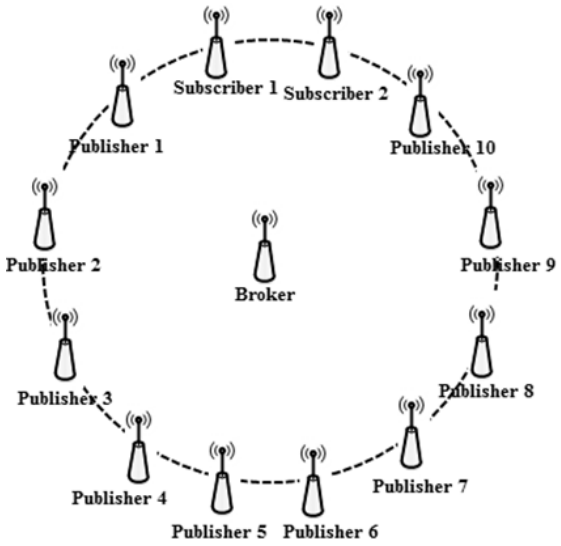


Figure 31. Single hop network topology

The extended single hop network topology consists of a distributed system based on multiple broker nodes, by using the Publish/Subscribe model as shown in Figure 32. This allows the extension of the coverage area of the application and the communication of interested parties (sensors and actuators) located in more than one hop distance of each other. Publication messages originated in publisher nodes located from more than one hop away are received through the broker node to which the subscriber nodes are connected. This is possible because of the broker node intercommunication. That is, broker node subscribes on behalf of its subscriber nodes to another broker node that the publisher node with information of interest connected to.

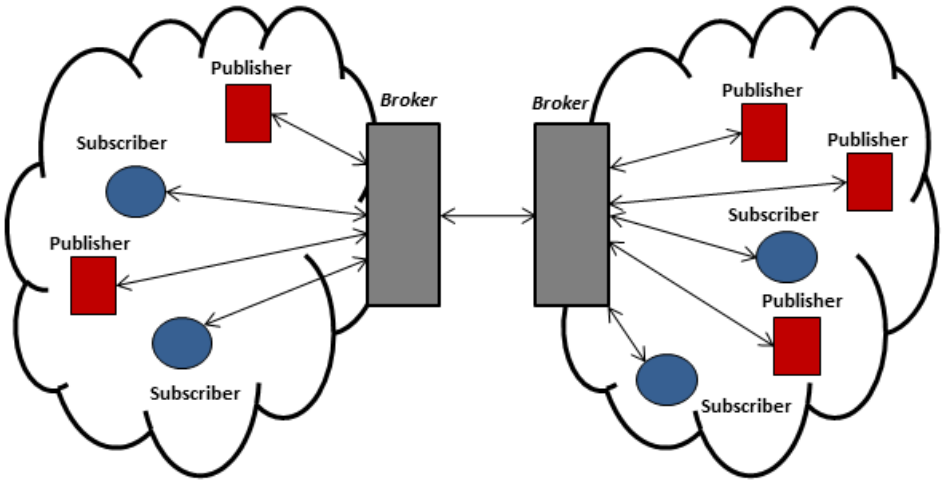


Figure 32. Single-Hop Extended Network Topology

On the other hand, the multi-hop scenario consists of nodes that are located up to 3 hops from the broker node, as can be seen in Figure 33. The messages originating from a publisher node are routed to the broker node through multiple nodes and the broker node has to route the messages in a similar way to the subscriber nodes. Static routes are defined for the sake of the simplicity of the simulation environments.

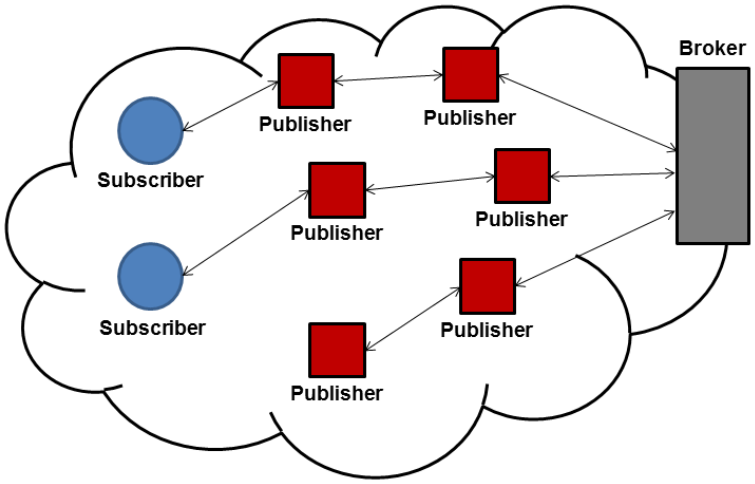


Figure 33. Multi-Hop Network Topology

In the PHY layer, we use the 2.4 GHz range with a bandwidth of 250 kbps based on IEEE 802.15.4 [31]. In addition, the maximum number of MAC-layer retransmissions is 3, which is the default value of IEEE 802.15.4 [31]. For energy consumption calculation, we use the energy model provided by the simulator [88] with the power consumption settings based on the TelosB datasheet [90].

Each simulation experiment lasts for 500 seconds. Each point in the graph presented in this section is based on the average of ten simulation runs. Table 4 summarizes the parameters and their assigned values used in the simulation.

Parameter	**Value**
Carrier Frequency	2.4 GHz
Bit rate	250 kbps
Max. number of retransmissions in App. Layer	4
Max. number of retransmissions in MAC layer	3 (default for IEEE 802.15.4)
Publication generation interval	1 second
Publication message size	74 bytes
Traffic type	Periodic

Table 4. Parameter setting for simulation

4.1.1 Performance Metrics

The packet delivery ratio, the dropped publication ratio, the retransmitted publication ratio and the duplicated publication ratio have been established as performance metrics in order to evaluate the reliability mechanism of MQTT-SN and CoAP protocols,

- **Packet Delivery Ratio (PDR):** This is a crucial metric for evaluating the performance of the reliability mechanism of MQTT-SN and CoAP protocols. It expresses the ratio of the total number of publication messages received by each subscriber node, up to the total number of publication messages generated by all publisher nodes of the events to which the subscriber node has subscribed. It does not consider duplicated publication messages received by subscriber nodes.

- **Discarded Publications Ratio (DPR):** With this metric we evaluate the impact of the publication discipline in the PDR. This is because the discarded publications will never be received by the subscriber node, so the PDR will decrease as the DPR increases. This measure is the ratio of the number of discarded publication messages (at the publisher nodes or at the broker), and the amount of messages generated (at the publisher nodes).

- **Retransmitted Publications Ratio:** It is the ratio of the total number of publication messages retransmitted to the total number of sent publications messages. This metric is evaluated for the total number of publisher nodes and for the broker node. This is an important metric used to evaluate the effect of the RTO value, because a good RTO value should reduce spurious retransmissions as well as ensuring reaction without delay in the case of message losses.

- **Duplicated Publications Ratio:** This indicates the ratio of the number of duplicated publication messages received to the total number of publication messages received. We evaluate this metric in the broker node and in the subscriber node with QoS1 (for MQTT-SN) and subscriber node with CON (for CoAP). It counts the number of retransmitted publications that reach the broker and the subscriber nodes.

Each of the above metrics was investigated by varying the number of publisher nodes and the K value used for RTO calculation, as previously explained. Furthermore, we investigate the effect of using or not using MAC Acknowledgements in order to find situations in which this mechanism may be needed.

4.2 Adaptive RTO mechanism

At the moment of this evaluation, CoAP presented a proposal considering an adaptive RTO calculation based on RFC 6298 [89] which is based on algorithm to compute a smoothed RTT (SRTT) and another algorithm to calculate an RTT variance (RTTVAR). All of results for CoAP protocol showed in this chapter, compare our proposal with CoAP using fixed RTO calculation (called default congestion control) and CoAP using RTO calculation based on RFC 6298. However, later the mechanism of adaptive RTO of CoAP

evolved with the name of CoCoA [23] and whose results will be compared with our proposal in the next chapter of this thesis.

In the case of RTO calculation for CoAP based on RFC 6298, we argue that this method is effective for obtaining a proper RTO value for the network conditions, although it is not effective for sensor nodes. The use of these two algorithms means that the state for each destination at the sender must be maintained, which may require high amount of resources in terms of memory and computing, which in turn results in more energy consumption in the sensor nodes.

We consider using the algorithm to obtain an estimated SRTT. Furthermore, in order to ensure a good estimation of RTO, we propose the multiplication of SRTT by a K factor, but not by using the RTTVAR. The RTO calculation is therefore performed by multiplying the estimated SRTT by a K parameter

$$RTO = SRTT \times K \tag{4.1}$$

where SRTT [89] is given by:

$$SRTT = (1 - \alpha) \times SRTT + \alpha \times RTT \tag{4.22}$$

where α = 1/8 and RTT is the time from when a publication message is generated, and the confirmation message is received at the application level.

This method would lead to a reduction in the process of computing RTO in the sensor nodes. Since nodes in WSN are constrained-resources devices, a lightweight computing would help to reduce energy waste and extend the lifetime of the node in the network.

Another situation in which the way of calculating the RTO plays an important role is in the publication discipline of MQTT-SN and CoAP protocols. The publication discipline of both protocols has a direct impact on the number of discarded publication messages. This is because the publication discipline will discard publication messages if a new publication messages is received while the RTO is activated.

It should be noted that publisher nodes always discard a greater number of publication messages in comparison with the broker node. This is because the publisher nodes are in charge of publication message generation. Thus, if a confirmation message is not received before a publication message is generated; the publication discipline discards the publication message.

The reasons why they have not received this confirmation could be:

- Because the broker has not received the publication message, thus requiring a retransmission from the publisher nodes.

- Because the broker has received the publication message, but the confirmation message is lost. Likewise, in this case the publication from the publisher could also be retransmitted, but it will become a duplicate publication for broker node.

In both cases, the discarded ratio of publications is proportional to the number of publisher nodes. In other words, as we increase the number of publishers, the discarded rate of publications increases. This is because a greater number of publisher nodes in the network may give rise to two situations: increased contention for access to the channel and a higher probability of collision, resulting in a greater delay in channel access. Both

situations cause application layer (CoAP or MQTT-SN) retransmissions and increase the probability of generating a new publication while waiting for a confirmation of a previous publication (RTO timer is activated).

As mentioned above, a fixed RTO value is unable to respond to changing network conditions thus, for larger values of RTO, the number of discarded publication messages could be high. As a consequence, the packet delivery ratio of subscriber nodes will also decrease.

The publication discipline of both protocols differs in the way it discards the publication messages. On one hand, MQTT-SN attempts a persistent delivery of a generated publication to the subscriber. This means that it discards a new publication if there is a pending confirmation of a previous publication. On the other hand, CoAP always attempts to deliver a new publication message. This means that a new publication is sent and the retransmission of the old publication is discarded (canceled) should it be needed.

Therefore, the effect of the RTO value on the publication discipline of MQTT-SN causes the discarding of new publication messages, which in turn results in the loss of new data in the subscriber node. In contrast, the effect on the publication discipline of CoAP is reflected in the cancellation of possible retransmissions of old publication messages with confirmable messages. As a consequence, the lost publication messages would be treated as a publication with a Non-Confirmable message, which in turn causes a decrease in the packet delivery ratio at the subscriber nodes. This is because MQTT-SN could be oriented to applications where the delivery of most of the data is required. In contrast, CoAP attempts to keep the subscriber node abreast of the most recent data from an event. The goodness of the MQTT-SN and CoAP publication discipline depends on the application area.

An adaptive RTO that takes network conditions into account would reduce the number of dropped publication messages for both protocols. However, the decrease in the number of discarded publications will also depend on the publication generation rate. This situation is beyond the control of an adaptive RTO.

4.3 Results and Discussion

In this Section, we analyze and discuss the results obtained from the evaluation of MQTT-SN and CoAP protocols by simulation using our adaptive method. We investigate the value of K from which we obtain the highest PDR for the subscriber node with QoS1 for MQTT-SN and the confirmable (CON) message for CoAP, respectively. Furthermore, in order to evaluate in what way, the adaptive RTO method is better than the one used by MQTT-SN and CoAP, we compare their PDR values. Finally, we evaluate the other metrics such as: discarded publication ratio, retransmitted publication ratio, duplicated publication ratio and energy consumption, in order to determine why the highest PDR is obtained with a specific K value.

4.3.1 Single Hop Scenario

In this section, we show and discuss the results obtained for the single hop that we have described previously.

4.3.1.1 Packet Delivery Ratio (PDR)

The PDR is affected by the number of publisher nodes. As the number of publisher nodes increases, a lower PDR is obtained. As expected, the probability of success in accessing the medium decreases when a greater number of nodes contend for access to

the channel. We evaluate the effect of the K value with our proposed method for RTO calculation, as well as the effect of using or not using MAC Acknowledgements. The aim of this evaluation is to find the appropriate K value for obtaining the highest PDR for subscriber nodes for MQTT-SN and CoAP.

In general, the subscriber nodes achieve a higher PDR as the K value increases for MQTT-SN and CoAP. One of the reasons for this is because spurious retransmissions are reduced. However, we found that for a value above a specific K value the PDR of subscriber nodes begins to decrease. On the other hand, when considering the effect of using or not using MAC Acknowledgements, both protocols show a similar behavior.

Firstly, for MQTT-SN, with our proposed method, we observe that the highest PDR for the subscriber node with QoS 0 is obtained with K=3, without the use of MAC Acknowledgements, as can be seen in Figure 34(a). However, Figure 34(b) shows that for a number of publisher nodes less than 40, the highest PDR is obtained by using MAC Acknowledgements with K =2.

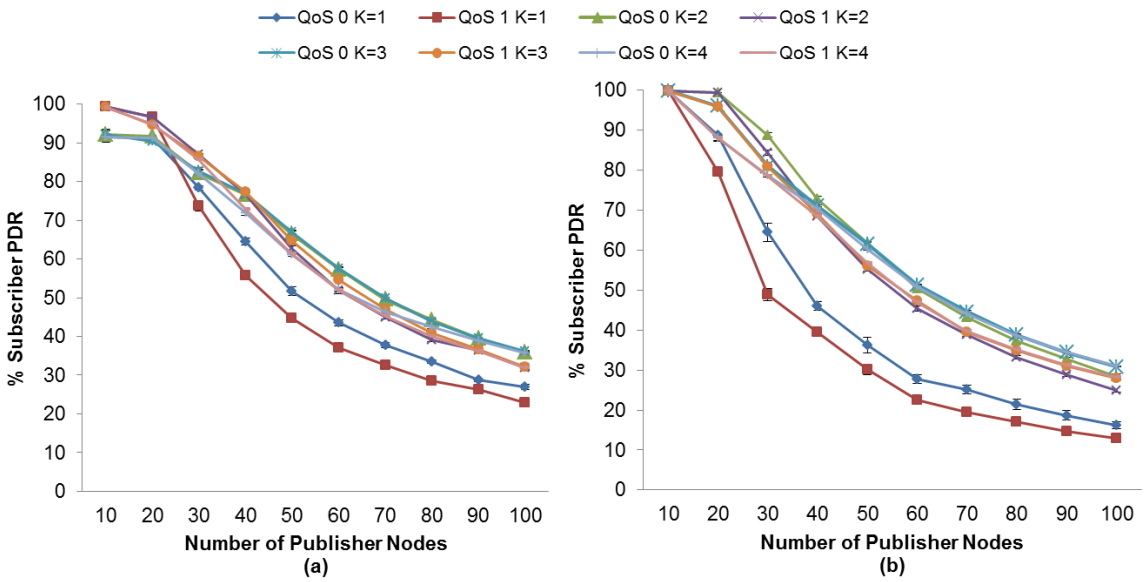


Figure 34. Effect of the number of Publisher Nodes on the Subscribers PDR depending of the K value of RTO with MQTT-SN (a) without and (b) with MAC Acknowledgements

In this case, the value of K=2 enables the MAC Acknowledgements to recover most of the lost messages before MQTT-SN retransmissions are activated from publisher nodes. Otherwise, a number of publisher nodes greater than 40 when using MAC Acknowledgements results in a higher probability of collision and loss of publication messages, and thus in an increase in delay. In this context, the value of K=2 would lead to spurious retransmissions because the MQTT-SN retransmissions would be activated before MAC Acknowledgements attempt to recover the lost messages. This situation results in a low PDR for this subscriber node.

A similar situation occurs for the subscriber node with QoS 1. The highest PDR is obtained with K=3 without the use of MAC Acknowledgements, as shown in Figure 4.4(a). An exception occurs for a number of publisher nodes less than 30; in this case the subscriber node with QoS 1 obtains the highest PDR employing K=2 with the use of MAC Acknowledgements, as shown in Figure 34(b).

Note that the PDR of a subscriber with QoS 1 changes from 30 publisher nodes onwards, unlike the case of the PDR of a subscriber with QoS 0, where the PDR changes from 40 publisher nodes onwards. This is because the additional messages used on the

reliability mechanism provided by QoS Level 1 of MQTT-SN congest faster with a number of nodes greater than 30 publisher nodes.

For CoAP, both subscriber nodes obtain the highest PDR with K=2, without the use of MAC Acknowledgements, as can be seen in Figure 35(a). Nevertheless, Figure 35(b) shows that for a number of publisher nodes less than 40, the use of MAC Acknowledgements is required for obtaining the highest PDR for a subscriber with NON messages. The same situation applies for the subscriber with CON messages. In this case, the use of MAC Acknowledgements is required for a number of publisher nodes less than 30 in order to obtain the highest PDR. The reasons for this are the same as those previously explained for the subscriber node with QoS 0 using the MQTT-SN protocol with our proposed RTO method.

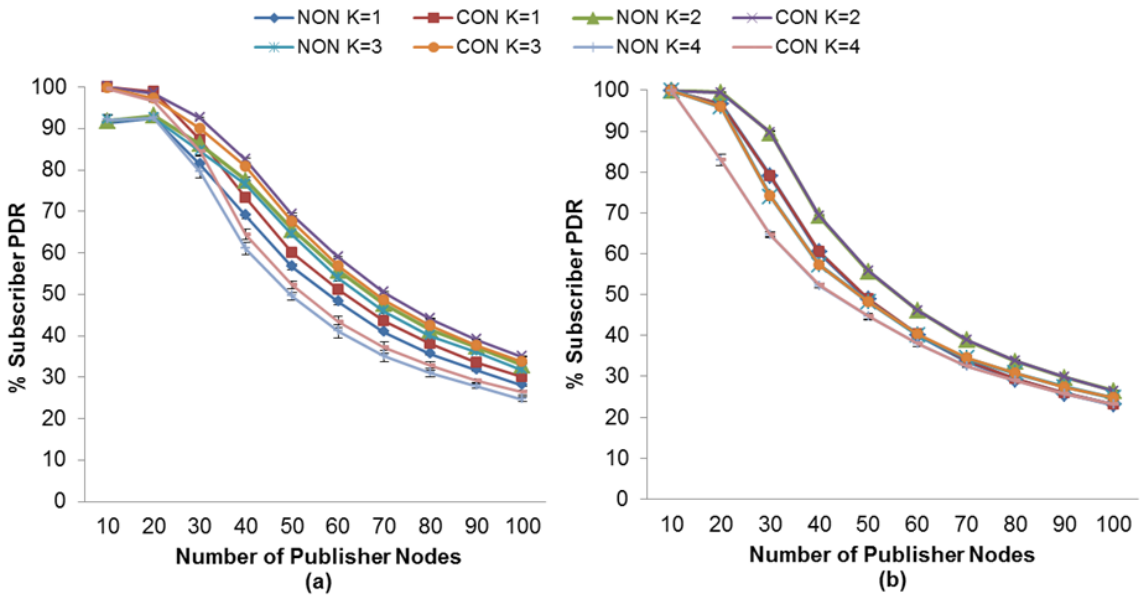


Figure 35. Effect of the number of Publisher Nodes on the Subscribers PDR depending of the K value of RTO with CoAP without (a) and with (b) MAC Acknowledgements

Another important aspect to consider in the PDR is the effect of the discarded publication ratio (DPR) caused by the publication discipline of each protocol, which is discussed in detail in the following section. For a higher DPR, a lower PDR for each subscriber node is obtained. For the publication discipline of the MQTT-SN protocol, a fraction of the publications generated by the publisher nodes or received by the broker node are discarded before being sent to the subscriber nodes. This fraction is not sent to the channel, and therefore reduces the number of publication messages received by the subscriber nodes. In the case of the CoAP publication discipline, neither the publisher nodes nor the broker node will retransmit a publication message pending confirmation if it generates or receives a new publication. This means that the publication messages will not be recovered in case of losses, and consequently the PDR of subscriber nodes is reduced.

In summary, one may observe that the publication discipline of CoAP leads to a PDR for the subscriber node with reliable delivery (QoS 1 and CON messages, respectively) that is higher than the publication discipline used by MQTT-SN. Recall that CoAP publication discipline gives priority to sending the new publications rather than attempting to retransmit the old one as MQTT-SN does. This situation increases the PDR

in the case where the publication messages have been received by the subscriber node but the confirmation is lost.

In addition, we compare the fixed RTO method used by MQTT-SN with our proposal. In Figure 36(a), one may observe that when MAC Acknowledgements are not used, subscriber nodes obtain a PDR with the fixed RTO method used by MQTT-SN that is lower than when using our proposal. As expected, the fixed RTO values of 10 and 15 seconds cause the MQTT-SN retransmissions to be activated too late to recover the message losses. The advantage of the RTO method we use is evident; the PDR for the subscriber node with QoS 0 obtains an increase in PDR of between 64% (for 20 publisher nodes) and 23% (100 publisher nodes) as compared with the fixed RTO method used by MQTT-SN. For the subscriber with QoS 1, this increase is between 76 % (for 10 publisher nodes) and 21% (for 100 publisher nodes). As explained earlier, our method considers the SRTT as a network condition metric in order for it to react properly when a message loss occurs.

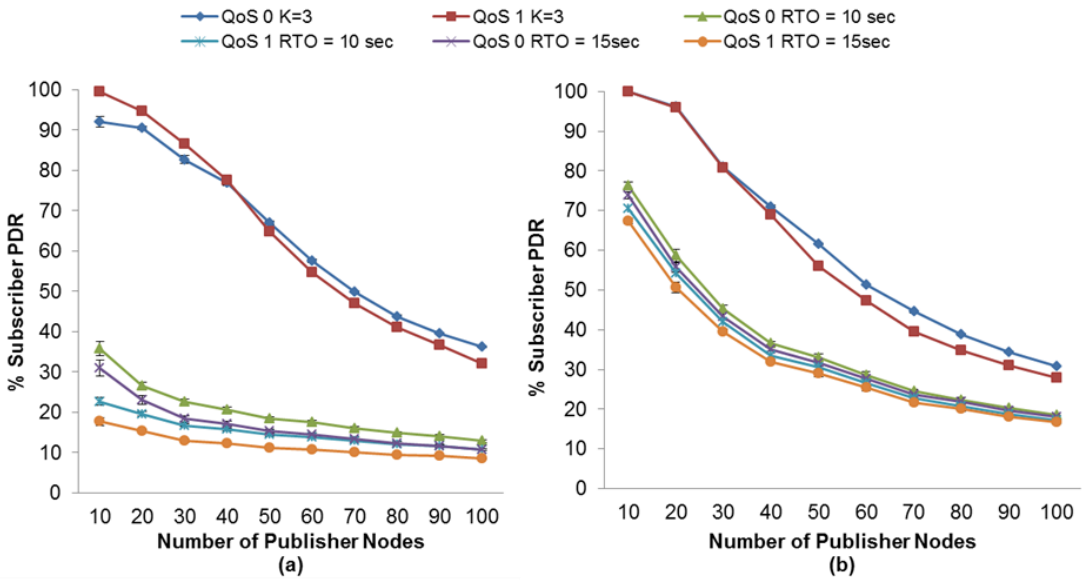


Figure 36. PDR comparison between the RTO method of MQTT-SN calculation and our proposal without (a) and with (b) MAC Acknowledgements

Although the MAC layer provides a reliability mechanism using MAC Acknowledgements, it is not sufficient to recover the message losses. As may be seen in Figure 36(b) below, although MAC Acknowledgements are used, the subscriber nodes with the RTO method used by MQTT-SN obtain a lower PDR than in our method.

Therefore, with the use of our adaptive RTO method, the subscriber with QoS 0 achieves an increase in its PDR of between 38% (for 20 publisher nodes) and 12% (for 100 publisher nodes). For the subscriber node with QoS 1, the PDR increase is between 40% (for 20 publisher nodes) and 10% (for 100 publisher nodes).

The same situation occurs when we compare the PDR employing the RTO method used by CoAP with our approach, as may be seen in Figure 37. Although the initial RTO of CoAP is selected between 2 and 3 seconds and is doubled for consecutive retransmissions, unlike in our approach it is not sufficient to obtain a higher PDR.

In Figure 37(a), one may observe that without the use of MAC Acknowledgements the subscriber node with NON messages obtains an increase in its PDR of between 34% (for 30 publisher nodes) and 13% (for 100 publisher nodes) using our adaptive RTO method.

In contrast, the increase in the PDR for a subscriber node with CON messages is between 38% (for 30 publisher nodes) and 14 % (for 100 publisher nodes). Moreover, when using MAC Acknowledgements, the PDR of both subscriber nodes increases between 26 % (for 30 publisher nodes) and 4% (for 100 publisher nodes), as shown in Figure 37(b).

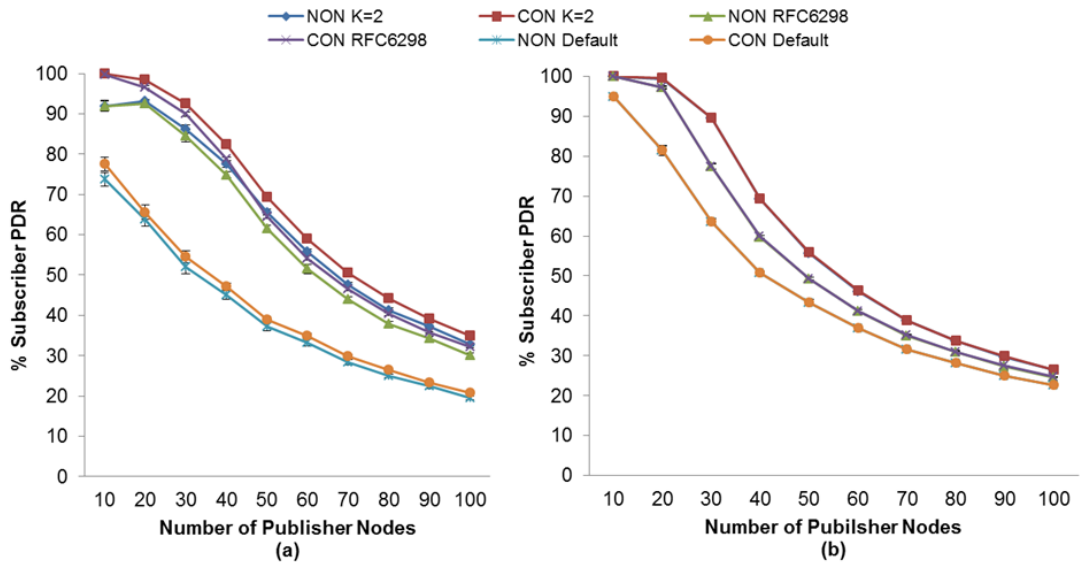


Figure 37. PDR comparisons between the RTO method of CoAP calculation and our proposal without (a) and with (b) MAC Acknowledgements

Moreover, comparing our adaptive RTO method with CoAP using RFC6298 one may observe that, as showed in Figure 37(a) without the use of MAC Acknowledgment, both subscriber nodes get an increase in its PDR of between 5% (for 50 publisher nodes) and 3% (for 100 publisher nodes) using our adaptive RTO method. Moreover, when using MAC Acknowledgements, the PDR of both subscriber nodes increases between 13 % (for 30 publisher nodes) and 1% (for 100 publisher nodes), as shown in Figure 37(b).

4.3.1.2 Comparison of RTT and RTO measurements

We compare the measured RTT with the RTO calculated with our method to gain an insight into the behavior of the RTO as regards RTT. For MQTT-SN, Figure 38(a) shows that without MAC Acknowledgements, the average RTT of publisher nodes is almost equal to that of the broker node, and in general the average RTO values are similar. The main difference occurs in the average RTO for publisher nodes, where this value is higher than for the broker node from 60 publisher nodes onwards. This is due to the fact that more publisher nodes are competing for access to the channel, and consequently the probability of packet collision increases. The RTO value therefore increases because the retransmissions from the application layer (MQTT-SN) are activated. A similar situation is depicted in Figure 38(b). However, in this case a higher RTO value is obtained due to the use of MAC Acknowledgements.

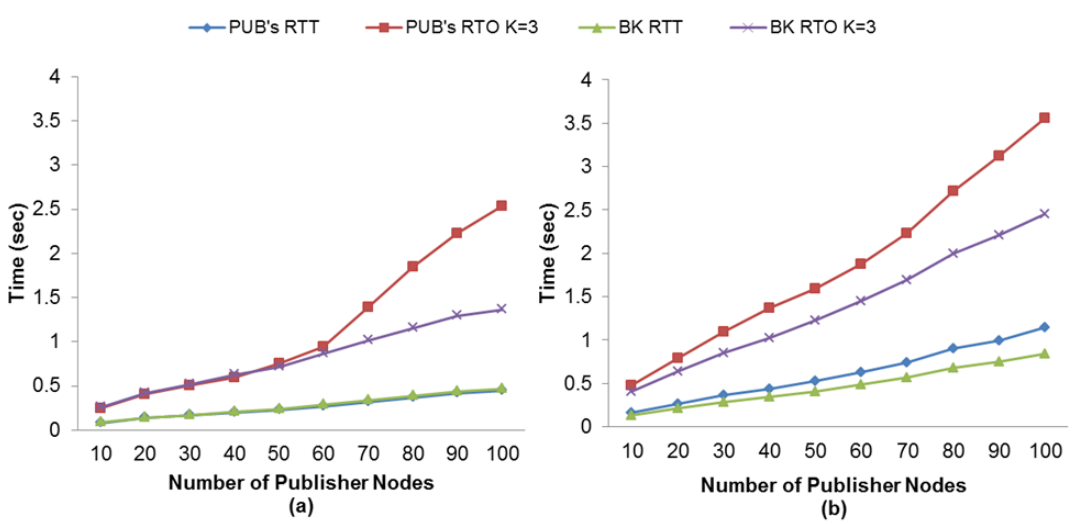


Figure 38. RTT and RTO comparisons for MQTT-SN with our method without (a) and with (b) the use of MAC Acknowledgements

On the other hand, Figure 39(a) shows the situation for CoAP without the use of MAC Acknowledgements. We can observe a change in the RTO behavior from 60 publisher nodes onward. The reason is the same as that previously explained for MQTT-SN. However, the RTO value increases very slowly due to publication discipline of CoAP, which in turn results in the cancellation of retransmissions from the application layer (CoAP).

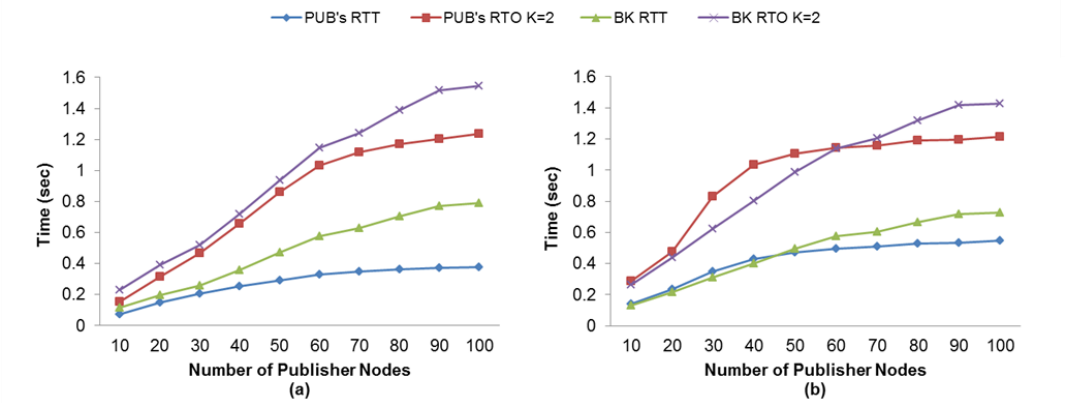


Figure 39. RTT and RTO comparisons for CoAP with our method without (a) and with (b) the use of MAC Acknowledgements

For the use of MAC Acknowledgements, the situation is the same, as illustrated in Figure 39(b); however, the change in the RTO behavior occurs from 40 publisher nodes onward. This difference is due to the fact that the use of MAC Acknowledgements causes the network to congest faster.

Comparison of the RTO for both protocols shows that, unlike MQTT-SN, in CoAP the increase in the RTO value is very low from 60 publisher nodes onward. This is mainly due to the publication discipline of CoAP. In summary, the obtained RTO value with our adaptive RTO method adapts better than the fixed RTO method used by MQTT-SN and CoAP.

4.3.1.3 Discarded Publication Ratio (DPR)

The publication discipline of MQTT-SN and CoAP has the effect of discarding publication messages, which in turn results in a reduction of the PDR. We evaluate the effect of the K value and the use of MAC Acknowledgements in the discarded publication ratio metric in order to find the setting with which we obtain the lowest DPR or match the highest PDR obtained in subscriber nodes. The results obtained show that for both protocols, MQTT-SN and CoAP, publisher nodes have a higher discarded publication ratio than the broker node. This may be observed in Figures 40 and 41 and is mainly due to the fact that the publisher nodes do not receive the confirmation message from the broker node because of the loss of the publication message, the loss of the confirmation message, or because there is no channel access due to channel congestion. As a consequence, publisher nodes will carry out unnecessary retransmissions, thereby leading to an increase in the duplicated publications in the broker node. Moreover, MQTT-SN will discard new publications, and CoAP will send a new publication and cancel possible retransmission of current one. The effect of retransmitted and duplicated messages is discussed in the following sections.

Regarding the effect of the K value, in both protocols we observe that DPR decreases in publisher nodes as the K value increases. However, our findings show that above a specific K value, the DPR increases. This specific K value depends on the protocol.

For MQTT-SN in general, the DPR also decreases as the K value increases, except for K = 4. In this case, the retransmission of MQTT-SN is activated too late to recover publication messages in the case of loss. This situation results in a higher DPR, since a new publication message could be generated while the RTO is activated, as can be seen in Figure 10. This situation occurs whether the MAC Acknowledgements is used or not.

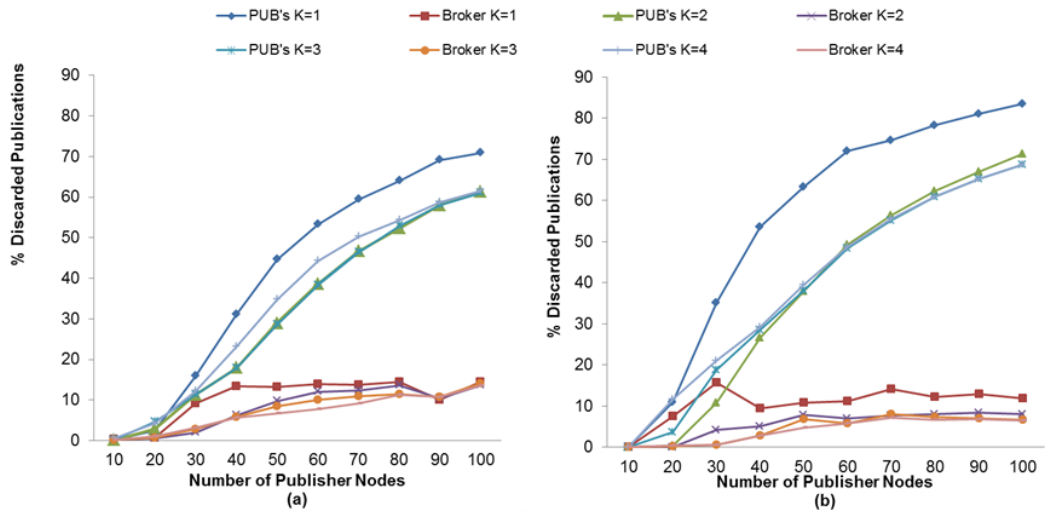


Figure 40. Effect of the number of Publisher nodes on the Discarded Publication Ratio from Broker and Publisher Nodes using different K values without (a) and with (b) MAC Acknowledgments

In general, the lowest DPR for MQTT-SN is obtained with K = 3 without the use of MAC Acknowledgements, as shown in Figure 40(a). However, Figure 40(b) shows that for a number of publisher nodes less than 40, we obtain the lowest DPR with K=2 and using MAC Acknowledgements.

This is due to the increase in the number of publisher nodes, which in turn results in collision and loss of messages. For a number of publisher nodes less than 40, the value of K=2 allows MAC Acknowledgements to recover most of the lost messages before MQTT-SN retransmissions are activated from publisher nodes, thus the probability of generating a new publication message while the RTO is activated is reduced.

In contrast, a number of publisher nodes greater than 40 using MAC Acknowledgements gives rise to a higher probability of collision and loss of messages, and therefore an increase in delay. In this context, the value of K=2 would lead to spurious retransmissions because the MQTT-SN retransmissions would be activated before MAC Acknowledgements attempt to recover the lost messages. This situation would result in the discarding of publication messages due to a higher probability of generating a new publication message while RTO is activated.

Use of the value K =3 enables most of the lost messages to be recovered before MQTT-SN retransmissions are activated. Therefore, the probability of generating a new publication message while RTO is activated is reduced and so is the DPR. Finally, the setting using K=3 matches the one that obtains the highest PDR for subscriber node with QoS 1, as seen previously in Figure 34(a).

A very similar situation occurs in the case of CoAP as that explained with MQTT-SN. In general, the publisher nodes obtain a lower DPR as the K value increases, except for values of K above K=2, as seen in Figure 41.

Regarding the use of MAC Acknowledgements, Figure 41(a) shows that the publisher nodes obtain a lower DPR without using MAC Acknowledgements than when using it; the absence of MAC Acknowledgements reduces the delay in receiving messages. As a consequence, the probability of generating a new publication is also reduced, and the RTO is activated. In fact, the setting for the lowest DPR is obtained without MAC Acknowledgements or K = 2, except for a number of publisher nodes less than 30, as shown in Figure 41(a). In this case, the lowest DPR is obtained with the use of MAC Acknowledgements, as can be seen in Figure 41(b). Moreover, this setting also matches the one that obtains the highest PDR for the subscriber node with CON messages, as seen previously in Figure 35(a).

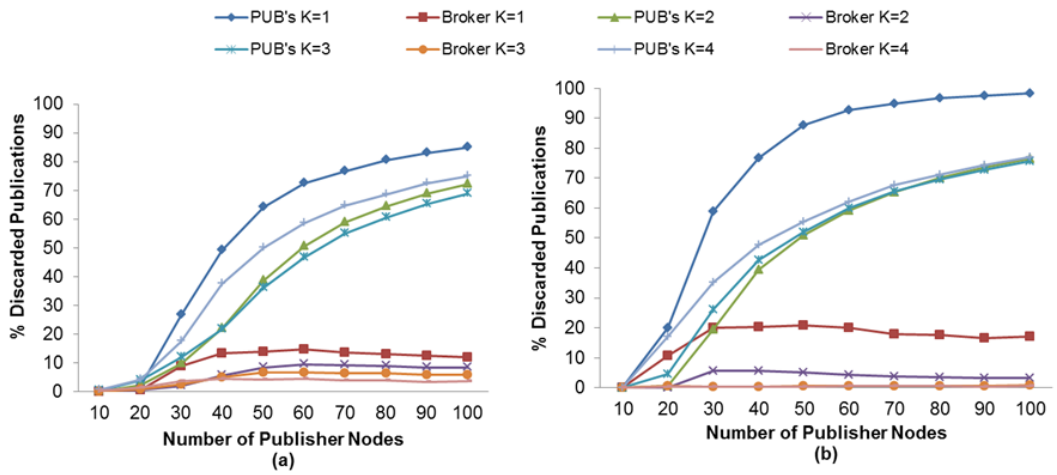


Figure 41 Effect of the number of Publisher nodes on the Dropped Publication Ratio from Broker and Publisher Nodes using different K values without (a) and with (b) MAC Acknowledgments

In summary, the highest DPR occurs in publisher nodes for both protocols, as pointed out previously. Nevertheless, we have observed that the CoAP protocol obtains a higher DPR than MQTT-SN protocol, based on the setting for the highest PDR of the subscriber node for each protocol. This situation is more evident for a number of publisher nodes greater than 30, as shown in Figures 40(a) and 41(a), and is due to the fact that an increase in the number of publisher nodes results in an increase in delay when sending the messages. Therefore, the probability of generating a new publication while the RTO is activated is greater, which results in the cancellation of the possible retransmission of the publication with a pending confirmation message and the transmission of the new one.

4.3.1.4 Retransmitted Publication Ratio

The RTO value plays an important role in the number of retransmitted messages. As previously mentioned, an unsuitable RTO value would give rise to spurious retransmissions as well as the inability to react in time to recover message losses, which in turn results in a lower PDR. We therefore evaluate the K value effect and the use of MAC Acknowledgments in the retransmitted publication ratio in order to find the setting with which to match the highest PDR obtained in subscriber nodes discussed in an earlier section.

Figures 42 and 43 show the retransmitted messages ratio for MQTT-SN and CoAP, respectively. It can be seen from these results that the lowest number of retransmitted messages is obtained with K= 4 with the use of MAC Acknowledgements. As expected, this is due to the decrease in the number of spurious retransmissions and the message recovery caused by the application layer retransmissions (MQTT-SN and CoAP respectively) and the use of MAC Acknowledgements, respectively.

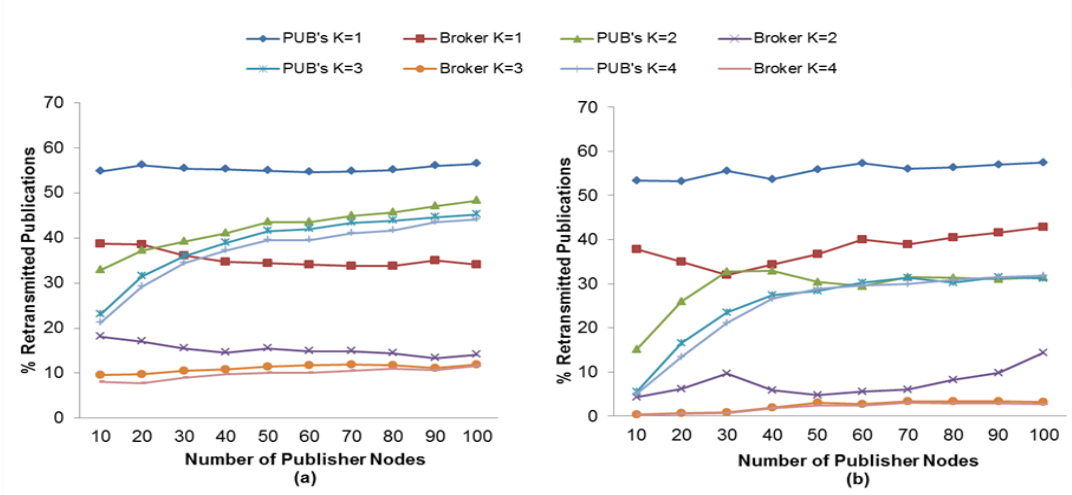


Figure 42. Effect of the number of Publisher nodes on publication messages retransmitted from Broker and Publisher Nodes using different K values without (a) and with (b) MAC Acknowledgments

Nevertheless, neither MQTT-SN nor CoAP obtain the highest PDR for subscriber nodes with this setting. As we have seen previously, for MQTT-SN and CoAP with QoS 1, and the subscriber node with CON messages, we obtain the highest PDR without the use of MAC Acknowledgements with K= 3 and with K=2, respectively. Comparison of both settings for MQTT-SN in Figure 42(a) shows that although the number of

retransmissions for K= 3 is higher than with K=4, as illustrated in Figure 42(b), these number of retransmissions are necessary to recover the publication message in order to obtain the highest PDR.

A similar situation occurs with the CoAP protocol, as can be observed in Figure 43(a) for K=2 and Figure 43(b) for K=4, respectively. Although the setting with K=4 enables the MAC layer to recover the publication message, this is insufficient.

For MQTT-SN, the retransmissions are activated too late for packet recovery due to a larger RTO value. The DPR will therefore increase, because a higher probability of generating a new publication message exists, and the RTO is activated. Hence, the new publication message will be dropped and the PDR will decrease.

In the case of CoAP, the same situation occurs for K=4. However, it should be pointed out that the decrease in PDR is due to the cancellation of possible retransmissions of publication messages. The publication messages with a confirmation pending will not be retransmitted in the case of loss, when a new publication is generated.

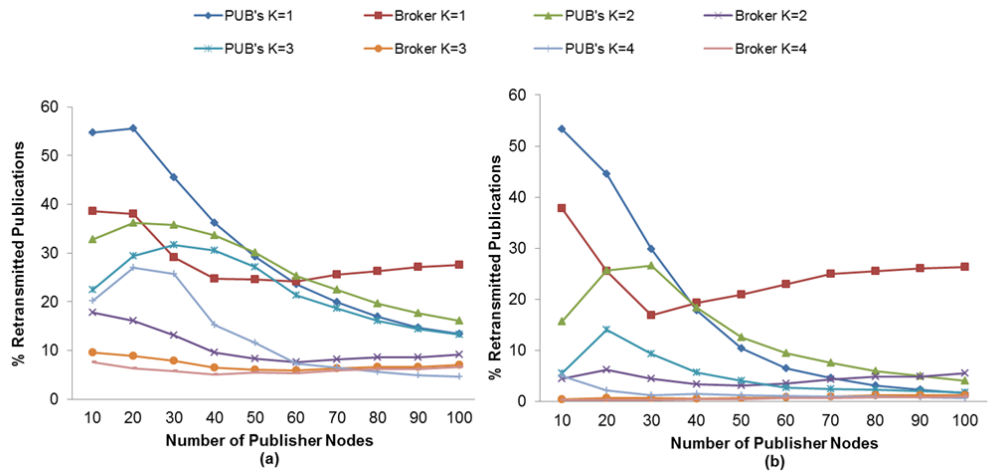


Figure 43. Effect of the number of Publisher nodes on publication messages retransmitted from Broker and Publisher Nodes using different K values without (a) and with (b) MAC Acknowledgments

Finally, a comparison of MQTT-SN with CoAP shows that the retransmitted message ratio for CoAP is lower than that for MQTT-SN, as one may observe in Figures 42(a) and 43(a), respectively. This is because the publication discipline of CoAP always will cancel the potential retransmissions of a current publication message when a new publication is generated. This situation becomes more evident as the number of publisher nodes increases (starting from 30 publisher nodes), since a larger RTO value is generated due to the message delay caused by channel contention or message loss, as can be seen in Figure 43(a).

In summary, we have seen that publication retransmissions have a direct impact on the PDR, depending on the K value for the RTO calculation and the publication discipline.

4.3.1.5 Duplicated Publications Ratio

Duplicated messages would reduce the PDR at the subscriber node because it receives useless data. In this context, we evaluate the effect of the K value and MAC Acknowledgment on the duplicated publication ratio to find the setting that matches the highest obtained PDR at subscriber nodes.

As expected, an inversely proportional relationship exists between the value of K and the number of received duplicated messages. As the value of K increases, the number of duplicated messages decreases, which in turn may cause a delayed reaction to packet recovery, especially when MAC Acknowledgements are not used. Otherwise, the number of duplicated messages increases due to spurious retransmissions. Both protocols show a similar behavior in terms of retransmitted messages.

In this context, Figures 44 and 45 show that both the subscriber nodes and the broker node have the lowest ratio of duplicated messages with K=4 and using MAC Acknowledgements.

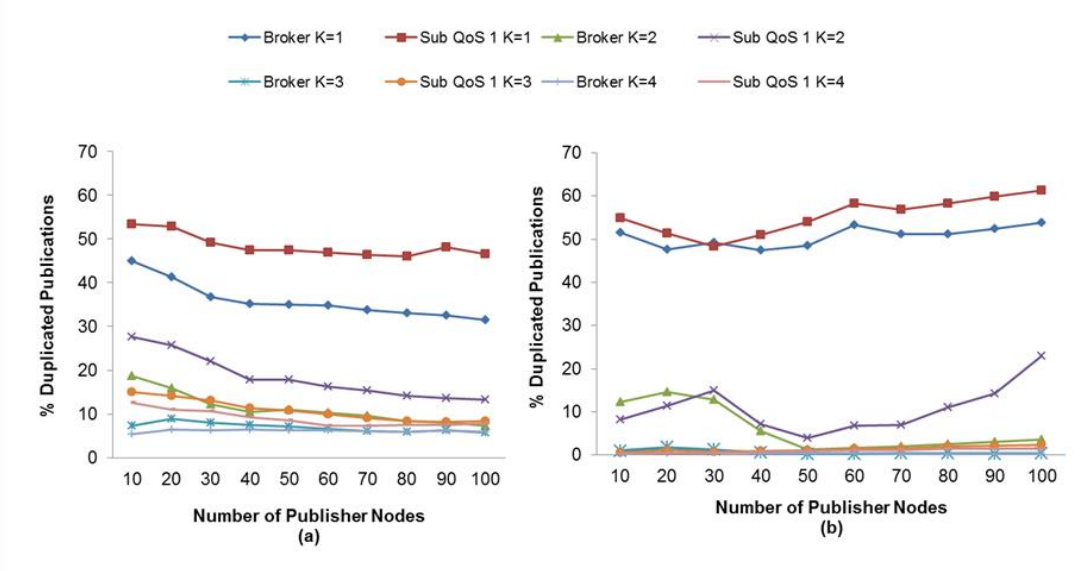


Figure 44. Effect of the number of Publisher nodes on the Duplicated Publications on Broker and Publisher Nodes depending on the K value of for RTO without (a) and with (b) MAC Acknowledgments

However, we have already seen that the highest PDR is obtained without the use of MAC Acknowledgements and with K=3 for MQTT-SN and with K=2 for CoAP, respectively. In this context, subscriber and broker nodes receive more duplicated publication messages, as shown in Figures 44(a) and 45(a). This is because MQTT-SN and CoAP react faster in the case of loss of a publication message, and consequently spurious retransmissions may be produced.

In effect, the results show that for the MQTT-SN protocol with K=3 without MAC Acknowledgements, most of the duplicated messages received in the broker node are caused by spurious retransmissions from the publisher nodes, as shown in Figure 44(a). From 20 publisher nodes upwards, most retransmissions are necessary for recovery of publication messages. In fact, for MQTT-SN, the duplicated message ratio decreases to approximately 6% from a number of publisher nodes greater than 20.

The situation is slightly different in the subscriber node with QoS 1. Figures 42(a) and 44(a) show that duplicated messages decrease to 8%, while retransmitted messages increase to 45%. One reason for this is that most of the duplicated messages are caused by spurious retransmissions from the broker node. The number of duplicated messages decreases proportionally as the number of publisher nodes increases in the network.

In the case of CoAP, most of the duplicated messages received in broker node are caused by spurious retransmission, as shown in Figure 45(a). Furthermore, the duplicated message ratio in broker node decreases as the retransmitted message ratio increases. This

is more evident from 60 publisher nodes upwards. The reason for this is that an increase in the number of publisher nodes in the network leads to a higher probability of collisions and an increment in the contention for channel access, a situation that generates a greater delay. This means the messages may be lost or sent with a delay, in which case the RTO from publisher nodes is activated.

Due to the CoAP publication discipline, the number of retransmitted messages will decrease because a larger RTO value leads to higher probability of generating a new publication message when the RTO is active. A similar situation occurs with the subscriber with CON messages.

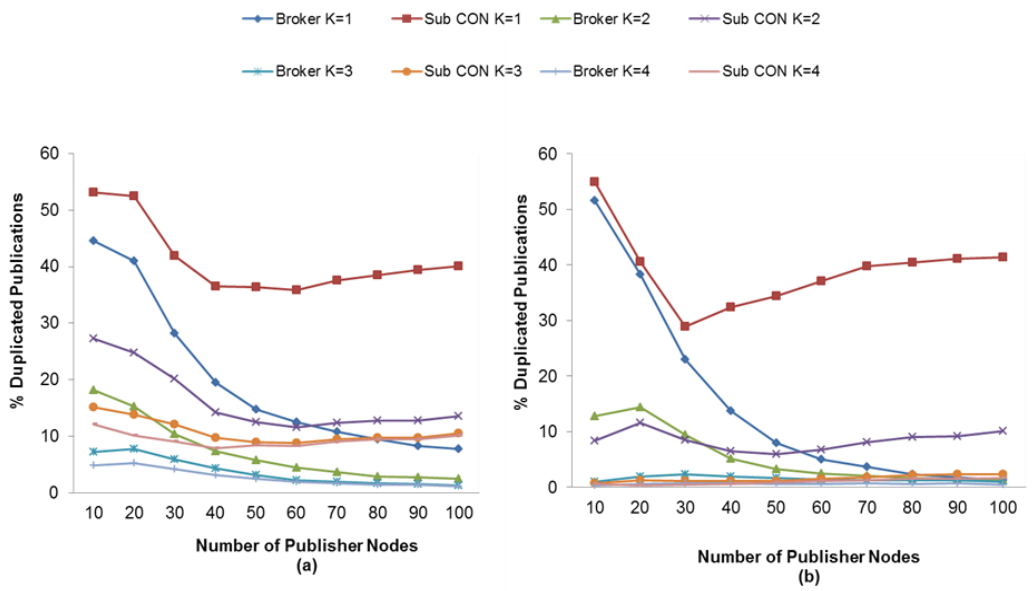


Figure 45. Effect of the number of Publisher nodes on the Duplicated Publications on Broker and Subscriber Nodes depending on the K value of for RTO without (a) and with (b) MAC Acknowledgments

Finally, one may observe that with CoAP we receive fewer duplicated messages than with MQTT-SN in the subscriber node with CON. The reason behind this situation is the difference in the publication discipline of both protocols that we discussed previously.

4.3.2 Single-Hop Extended Network Topology

For this scenario, we observe that the behavior of K value is still the same as above scenario. That is, for MQTT-SN the best value was K=3 and for CoAP this value was K=2 to obtain the highest PDR for subscriber nodes. Besides, as we expected, the obtained PDR was lower than the one in the single-hop scenario. For MQTT-SN the PDR for subscriber node with QoS 0 was 2% (for 10 nodes) and 22% (40 nodes) lower than single-hop scenario. For the subscriber node with QoS 1 the PDR was between 2% (10 nodes) and 24% (40 nodes) lower. For CoAP the PDR for subscriber with NON messages was up to 2% (10 nodes) and 26 % (40 nodes) lower than the single-hop scenario. Besides, the subscriber with CON messages gets a PDR 2% (10 nodes) and 24% (40 nodes) lower. The reason of that is because the addition of a second broker increases the network load which in turn results in packet losses. Besides, the messages between the broker nodes are sent in QoS 1. Therefore, for one of the broker nodes, the other broker node behaves

as another subscriber with QoS 1. This situation results in more congestion, thus increasing of packet losses. Furthermore, the network load obtained with 70 publisher nodes is similar to the one obtained for 100 publisher nodes in single hop scenario. This is due to the increase network traffic caused by the second broker node. For this reason, the result is showed up to 70 publisher nodes.

However, regarding the use of MAC Acknowledgment, the results show different behavior depending on the protocol. In this context, for MQTT-SN, the subscriber node with QoS 0 gets a higher PDR with the use of MAC Acknowledgment as shown in Figure 46(a). This is because the MAC Acknowledgments allow recovering the most of packet losses. In contrast, the subscriber node with QoS 1 gets the highest PDR without the use of MAC Acknowledgment. Nevertheless, Figure 46(b) shows that for a number of publisher nodes less than 40, the use of MAC Acknowledgment is required. The reason of this situation is because MAC Acknowledgment can recover the lost packets in situations of low traffic (up to 40 nodes). However, after this number of publisher nodes, the MAC Acknowledgments would congest the network faster resulting in an increase in packet delay and also in packet losses. This situation leads to an increase in probability to receive a new publication message from the application layer, while waiting for the ACK of an already sent one, which results in higher number of discarded publications. In this situation, the reliability mechanism of MQTT-SN with K=3 is the best among the ones evaluated to recover from packet losses without MAC Acknowledgment to get the highest PDR.

On the other hand, comparing the PDR obtained with our adaptive RTO method and the fixed RTO method we see that in MQTT-SN, for the subscriber node with QoS 0, the PDR increase is between 69% (10 nodes) and 26% (70 nodes). In the case of subscriber node with QoS 1, the PDR increase is between 71 % (10 nodes) and 27% (70 nodes). This demonstrates that the MQTT-SN fixed RTO method is not suited to react to packet losses in this situation.

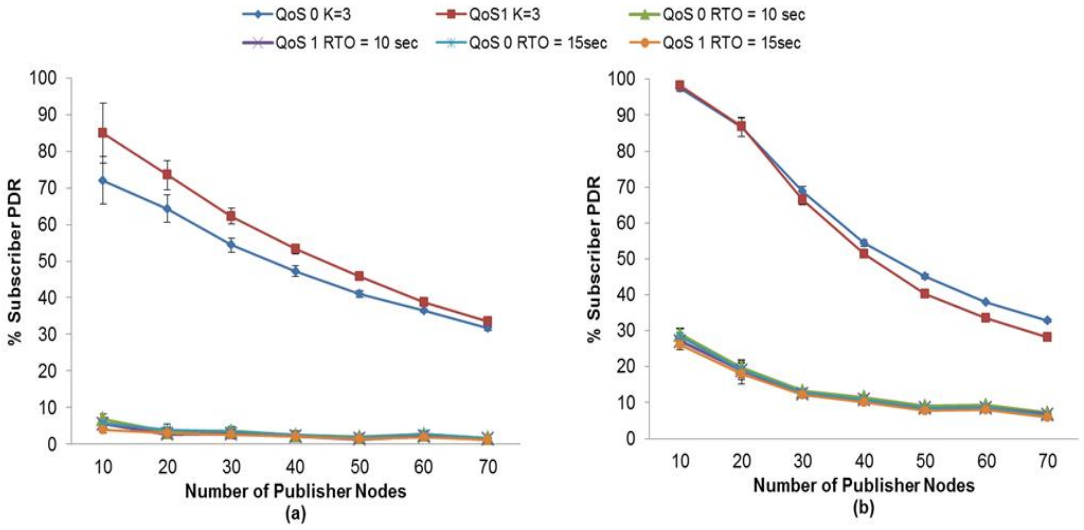


Figure 46. PDR comparisons between the RTO method of MQTT-SN calculation and our proposal without (a) and with (b) MAC Acknowledgements

For CoAP, we found that both subscriber nodes obtain the highest PDR without MAC Acknowledgements, as shown in Figure 18(a). Nevertheless, Figure 47(b) shows that for a number of publisher nodes less than 40, the use of MAC Acknowledgements is required for obtaining the highest PDR for a subscriber with NON messages. A similar situation

applies for the subscriber with CON messages. In this case, the use of MAC Acknowledgements is required for a number of publisher nodes less than 30 in order to obtain the highest PDR. The reasons for this are the same as those previously explained for the subscriber node with QoS 1 using the MQTT-SN protocol with our proposed RTO method.

On the other hand, the benefits for the subscriber node with NON messages using our RTO method is between 25 % (20 nodes) and 4% (70 nodes) more than using the CoAP fixed RTO method as can be seen in Figure 47(a). In the case of subscriber node with CON messages this increase is between 25% (10nodes) and 9% (70nodes) shown in Figure 47(a). Besides, comparing CoAP using our RTO adaptive method and the one using RFC 6298, we observe that subscriber node with NON messages gets up to 2% more PDR using our adaptive RTO method. For the same comparison, the subscriber node with CON messages, the increase of PDR is around 5% as can be seen in Figure 47(b). The reason of this is that using RFC 6298 RTO method the publisher nodes discards between 2% to 5% more publication messages than our adaptive RTO method. Besides, the duplicated messages ratio increases around 3% to 10 % more than our adaptive RTO method.

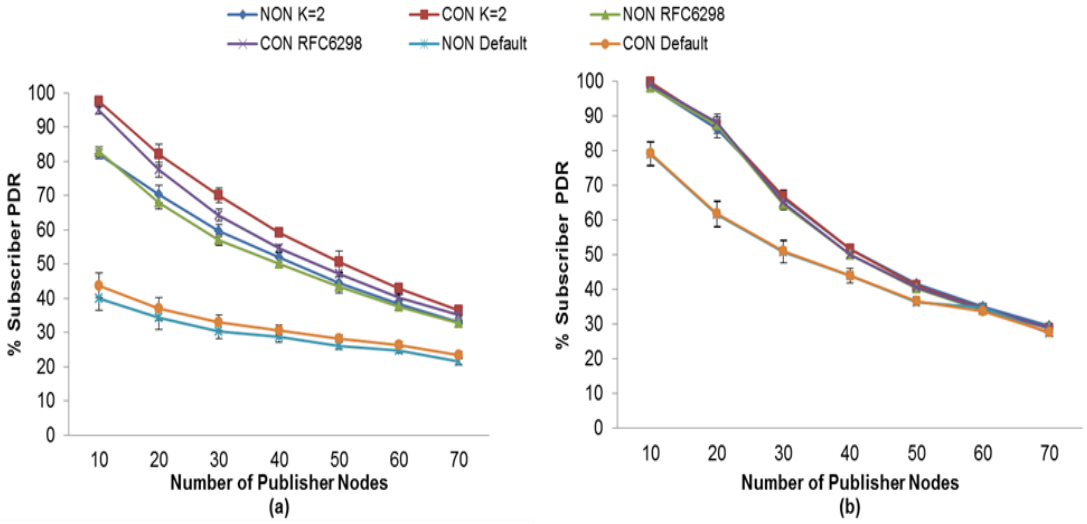


Figure 47. PDR comparisons between the RTO method of MQTT-SN calculation and our proposal without (a) and with (b) MAC Acknowledgements

In the case of the retransmission publication ratio, this ratio decreases as the K value decreases. With K=4 we get the lowest retransmission publication ratio and with use of MAC Acknowledgment. As expected, this is due to the application layer retransmissions (MQTT-SN and CoAP respectively) and the use of MAC Acknowledgements decreases the number of spurious retransmissions and the message recovery. However, with this value the PDR for MQTT-SN and CoAP decreases. This situation demonstrates that although the number of retransmissions for K= 3 for MQTT-SN and K=2 for CoAP is higher than with K=4, these number of retransmissions are necessary to recover the publication message in order to obtain the highest PDR. The K value for the duplicated message ratio and discarded publication ratio (DPR) shows the same behavior than the retransmitted publication ratio we have explained.

Finally, we have studied the relation of the energy consumption of the nodes with the K value. As we expected, as the K value increases, the nodes energy consumption is

decreased. For both protocols, the lowest energy consumption of the nodes is obtained with K=4 and we also get the lowest retransmitted publication ratio. However, for this K value we obtain a lower PDR. That is, for MQTT-SN the PDR decreases up to 5% for subscriber with QoS 0 and for subscriber with QoS 1 obtains up to 3% less PDR. In case of CoAP, the subscriber node with NON messages gets up to10% less PDR and for subscriber node with CON messages this decrease is up to 5%.

Moreover, we have compared the energy consumption of our adaptive RTO method with the MQTT-SN and CoAP. The results showed that with the use of the RTO methods of MQTT-SN and CoAP the nodes consume up to 8% less energy compared with our RTO method, which creates a trade-off between energy consumption and the PDR.

Based on an overall comparison between MQTT-SN and CoAP approaches, we see that the maximum achieved PDR by CoAP is better than that is achieved by MQTT-SN. This is related to the publication discipline used. If the PDR of reliable node's delivery is the objective of an application our findings propose the use of CoAP.

4.3.3 Multi-Hop Scenario

For this scenario, the highest PDR for nodes was obtained using different K values compared with previous scenarios for both protocols. Besides, we can see that for both protocols the network load with 40 publisher nodes is very similar to the one with 100 publisher nodes in single-hop scenario. Therefore, we have obtained the results up to 40 publisher nodes.

For MQTT-SN both subscriber nodes get the highest PDR with value of K=3.5 and using MAC Acknowledgments as shown in Figure 48(b).

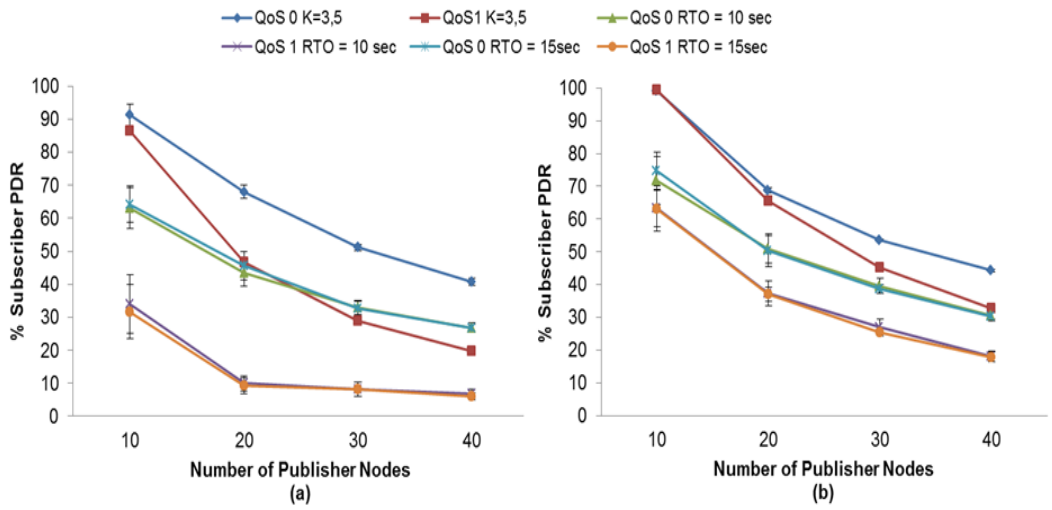


Figure 48. PDR comparisons between the RTO method of MQTT-SN calculation and our proposal without (a) and with (b) MAC Acknowledgements

The reason is the increase in RTT and the increase in packet losses caused by the different link conditions on each hop in the route to the destination. Therefore, the use of MAC Acknowledgment is necessary to recover most of the packet losses on each hop. Moreover, the value of K=3.5 is proper to react to packet losses in situations where MAC Acknowledgments are not sufficient to recover from packet losses.

In the case of CoAP, with K=2.5 both subscriber nodes get the highest PDR without the use of MAC Acknowledgments as can be seen in Figure 49(a). The reason is that without the use of MAC Acknowledgments, the reliability mechanism of CoAP can react

properly in case of packet losses for this K value. Otherwise, the use of MAC Acknowledgements leads to an increase of the message delay. In this situation, the value of K=2.5 would result in spurious retransmissions because the CoAP retransmissions would be activated before MAC Acknowledgements attempt to recover the lost messages. This situation results in a low PDR for this subscriber node.

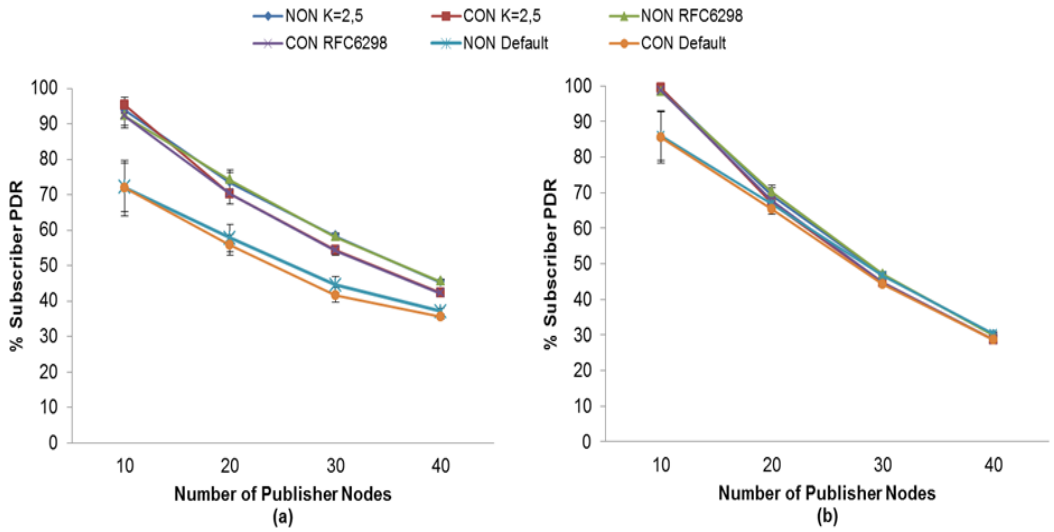


Figure 49. PDR comparisons between the RTO method of CoAP calculation and our proposal without (a) and with (b) MAC Acknowledgements

We also compared the PDR obtained with our adaptive RTO method and the fixed RTO method. The results show that for MQTT-SN, the subscriber node with QoS 0, the PDR increase is between 27% (10 nodes) and 14% (40 nodes). For subscriber node with QoS 1, this increase is between 15% (10 nodes) and 36 % (40 nodes). This demonstrates that the MQTT-SN fixed RTO method using a value of with 10 and 15 seconds is not suited to react to packet losses in this scenario.

For CoAP, the benefits for the subscriber node with NON messages using our RTO method is between 7% (20 nodes) and 15% (40 nodes) increase in PDR compared to the fixed RTO method used by CoAP. The PDR for the subscriber node with CON messages obtains an increase of between 5% (20 nodes) and 14% (40 nodes) as compared with the fixed RTO method used by CoAP.

Besides, comparing CoAP using our RTO adaptive method and the one using RFC 6298, we observe that subscriber node with NON messages gets up to 2% more PDR using our adaptive RTO method. For the same comparison, the subscriber node with CON messages, the PDR increase is around 5%. The reason of this is that using RFC 6298 RTO method the publisher nodes discards between 2% to 5% more publication messages than our adaptive RTO method. Besides the duplicated messages ratio increases around 3% to 10 % more than our adaptive RTO method.

For the other calculated metrics, we discuss the most important results to justify the reason because the subscriber nodes of both protocols get the highest PDR with K values, respectively. In the case of retransmitted publication ratio, the relation between the K value and the retransmitted publication ratio is the same as we have explained for previous scenarios. However, for MQTT-SN, we get the lowest ratio with the use of MAC Acknowledgments, this is the reason because the subscriber nodes get the highest PDR

with K=3.5. In the case of CoAP, although the retransmitted publication ratio is lowest with the use of MAC Acknowledgments this is not very relevant to get the highest PDR. On the other hand, for discarded publication ratio (DPR) the results show that for CoAP, with K=2.5 we get the lowest DPR on publisher nodes without the use of MAC Acknowledgments. This is because the subscriber nodes get the highest PDR without MAC Acknowledgments. In the case of MQTT-SN, we get a lower DPR without MAC Acknowledgments, but this is not very significant to get the highest PDR.

Regarding the energy consumption, the results show the same behavior as in the previous scenario. That is, the energy consumption increases with the increase of retransmitted publication ratio. Regarding the energy consumption of our adaptive RTO method compared with the MQTT-SN and CoAP, we find that the nodes consume in this scenario up to 6% less energy compared with our method, while losing from the PDR performance. It can be noted that our RTO method consumes only 1% more energy than the one on RFC 6298 used by CoAP.

Moreover, as we have showed for the studied scenarios, with all of these methods, the subscriber nodes achieve less PDR than our adaptive RTO method. This situation results in a trade-off between PDR and energy consumption. Therefore, we suggest that in applications in which PDR is not a critical requirement but the energy saving is very important, the choice of the RTO method should be considered.

Based on an overall comparison between MQTT-SN and CoAP approaches, we observe that CoAP gets the maximum achieved PDR better than that is achieved by MQTT-SN. This is related to the maximum achieved PDR by CoAP is better than that is achieved by MQTT-SN. This is related to the publication discipline used. Our findings propose the use of CoAP in case of the PDR of reliable node's delivery is the objective of the application.

Finally considering the metrics we have discussed in this section we could design a mechanism to adapt the K value to the network conditions. The broker nodes could inform to publisher and subscriber nodes the K value to be used through piggybacked information in the confirmation or publication messages. The broker node begins with an initial K value, which can be adapted depending on duplicated publication ratio, DPR and retransmitted publication ratio we have discussed previously. Also, the broker should consider the receiving message rate, the number of publisher nodes to calculate the network load and also the number of hops to the destination.

4.4 Conclusion and Contributions

In this chapter, we have presented and discussed a new adaptive RTO calculation method more suitable to react properly to changing network conditions compared with the proposed mechanism by MQTT-SN and CoAP. In this method we have considered the use of SRTT measurement and a K factor.

We evaluate three different scenarios: single-hop, single-hop extended and multi-hop scenarios, for which we perform simulations for the RTO methods used by MQTT-SN, CoAP and our proposal, along with the MAC Acknowledgment option for ensuring the one hop packet delivery.

The results show that for the single hop scenario the adaptive RTO method we use provides an increase in PDR of between 64% (for 20 publisher nodes) and 23% (100 publisher nodes) for the subscriber node with QoS 0 as against the fixed RTO method used by MQTT-SN. For the subscriber with QoS 1, this increase is between 76% (for 10 publisher nodes) and 21% (for 100 publisher nodes). These results are obtained without the use of MAC Acknowledgements in either method. Furthermore, the adaptive RTO

method using MAC Acknowledgements also is higher than the fixed RTO method for both protocols. In this context, for MQTT-SN, the subscriber with QoS 0 obtains an increase of between 38% (for 20 publisher nodes) and 12% (for 100 publisher nodes). For subscriber nodes with QoS 1, the PDR increase is between 40% (for 20 publisher nodes) and 10% (for 100 publisher nodes).

For CoAP, on the other hand, without the use of MAC Acknowledgements, the subscriber node with NON obtains an increase in its PDR of between 34% (for 30 publisher nodes) and 13% (for 100 publisher nodes). In contrast, the increase in the PDR for the subscriber node with CON is between 38% (for 30 publisher nodes) and 14% (for 100 publisher nodes). In addition, with the use of MAC Acknowledgements, the PDR of both subscriber nodes shows an increase of between 26% (for 30 publisher nodes) and 4% (for 100 publisher nodes).

For the multi-hop scenario, the results show that our adaptive RTO method still provides a higher PDR for subscriber nodes compared to the other RTO methods. For MQTT-SN, the increase of PDR is up to 36% for the subscriber node with reliability communication (QoS1 or CON message). Meanwhile, for CoAP this increase is up to 14% also for the subscriber node with reliability communication.

The results showed that our adaptive RTO mechanism provides an increase in PDR for each subscriber node compared with MQTT-SN and CoAP. Moreover, the results show the effect of the chosen K value is mainly on the packet delivery ratio and the discarded publication ratio. We find the setting for obtaining the highest PDR for subscriber nodes, mainly for the node receiving publication messages in reliable mode (QoS 1 and CON). In general, an increase in the K value yields a higher PDR and a lower discarded publication ratio. In fact, the highest PDR using our adaptive RTO is obtained with K = 3 and 2 for MQTT-SN and CoAP, respectively for single hop scenario.

Meanwhile for multi-hop scenario the value found was K=3.5 and 2.5 for MQTT-and CoAP, respectively. This behavior of K value demonstrates that we can obtain an optimized K value for each scenario to adapt to network conditions. However, it is also necessary to consider that this could lead to spurious retransmissions, and thus to duplicated messages.

In addition, we identify the situations in which MAC Acknowledgements are useful for packet recovery, and consequently obtain a reduction in application layer retransmissions. We conclude that, for the lowest evaluated K value, the use of MAC Acknowledgements is not recommended because it may give rise to high network congestion and consequently to a decrease in the PDR of the subscriber nodes.

Finally, regarding the publication discipline of each protocol, we also conclude that the non-persistent mode used by CoAP leads to a higher PDR than that in the persistent mode used by MQTT-SN. This is due to the fact that the CoAP publication discipline gives priority to sending the new publications while MQTT-SN attempts to retransmit the old ones, a situation which increases PDR when the publication messages are received by the subscriber node, but the confirmation is lost. The choice of the publication discipline depends on the application area.

Evaluation of the results shows that the RTO calculation plays an important role in all the metrics evaluated. There is a trade-off between PDR and the retransmitted message ratio, depending on the K value. Therefore, the investigation demonstrates that we can achieve better performance by using the optimized K value.

The results of the research presented in this chapter led to the publication of the article *Improving Packet Delivery Performance of Publish/Subscribe Protocols in Wireless Sensor Networks in* Sensors *[P1]*.

5. QoS Provisioning for Publish/Subscribe Protocols

In addition to the reliability in the delivery of packets that require applications to have the ability to respond appropriately to an event, another aspect to consider is the timeliness that refers to the guarantee of on-time delivery of packets. These are important aspects to consider for the publish/subscribe protocols in the WSN so that they can be a feasible solution in application scenarios such as smart city [91], [92] and e-health [93].

We consider the publish/subscribe model could provide a smart way to monitor and maintain the functioning and operation of the critical infrastructures of smart city such as gas, water, and electric systems which could save maintenance costs, repairs and ensure the wellbeing of the populace.

For example, we consider a scenario where wireless sensors are deployed to monitor electrical substation environmental parameters as temperature and humidity or physical security. The sensors are attached to different places such as power transformer, power lines and so on.

There are different types of messages such as normal operation, alarm and critical. The application needs to meet different QoS requirements for each of these messages. For example, a normal operation message should be delivered without reliability since it does not require performing an action; this could be the case of normal temperature values from the power transformer. However, if the temperature of substation is too high, the power equipment cannot work normally. In this situation, alarm message should be transmitted in order to control the air-conditioning, exhaust fan automatically according to the temperature. Thus, it is crucial that sensor networks provide a reliable delivery of every alarm message back to the application in order to register abnormal conditions and to perform a proper action.

On the other hand, we could consider electrical substations with several movement sensors used to detect unauthorized intrusions. This sensed data has a deadline, that is, it needs to be delivered within a time-limited bound in order to react properly to risky situations in real-time. This would be a use case for critical messages. In this sense, there are few proposals that focus on this issue and they satisfy this requirement only through priority mechanisms in which a packet with a defined deadline is sent first than the rest of the others. These proposals do not consider whether the deadline of packets has expired before reaching destination.

In the above realistic scenario, the Publish/Subscribe model is a feasible communication model since there are a large number of nodes that publish data from its different environmental sensors (publisher nodes) and a small number of nodes that are interested to receive and consume this data (subscriber nodes).

Considering the mentioned above application scenario and the presented QoS requirements application as a motivation, in this chapter we presented the discussion and results of a proposal of a mechanism that establishes different QoS levels to providing mainly reliability and timeliness on packet delivery based on Publish/Subscribe model for wireless sensor networks to meet application requirements. In addition, our proposal includes the use

of data aggregation with the reliability of packet delivery in order to be efficient in terms of energy consumption and the use of network bandwidth

5.1 QoS Levels Proposal

In this section, we describe each one of the QoS levels we proposed to meet the requirements regarding to reliability and timeliness of delivery packet. We propose three QoS levels: QoS level 1 provides reliability of packet delivery based on enhancement of our adaptive RTO mechanism presented on previous chapter, QoS level 2 adds the use of data aggregation with the aim to be efficient in energy and network bandwidth consumption and QoS level 3 provides timeliness by using a deadline mechanism.

5.1.1 QoS Level 1 (Reliable Packet Delivery)

Packets losses occur due to the specific nature of the wireless links, or network congestion caused by multiple nodes attempting to transmit its data. As we have discussed above, several protocols propose different mechanisms to determine the appropriate retransmission timeout (RTO) to consider that a packet is lost and transmit it again. Aspects such as number of duplicate received packets, number of retransmitted packets, and number of received packets should be considered when designing an RTO mechanism because that could result in increasing resources consumption in an unnecessary manner such as: network bandwidth, energy, processing, and the decrement of the Packet Delivery Ratio (PDR).

To provide a reliable delivery packet, we propose a new approach to determine the RTO taking into account the aforementioned important aspects about RTO design to use efficiently the resources of the WSN. Specifically, we will use the RTO mechanism that uses a K parameter to calculate the RTO presented in the previous chapter. We recall we proposed to use the same algorithm to compute the RTO value in the same way as RFC 6298 but not using the Round-Trip Time variance (RTTVAR), instead of this, we proposed the multiplication of Smoothed RTT (SRTT) by a K factor. In previous works, we studied the effect of different fixed K values to improve the PDR in the subscriber nodes. In the present work, we adjust the K value in a dynamic way by using a feedback information from the subscriber nodes.

5.1.1.1 Proposed adaptive RTO mechanism

Commonly, the RTT parameter is generally used to estimate the condition of the network in the calculation of RTO. Also, when the RTO expires generally it assumes that the data packet was lost, however it does not take into account other causes such as the loss of the confirmation packet or unexpected delays. This absence of further information could result in an unsuitable calculation of RTO and it does not reflect other cases or situations. We could adjust the value of RTO for subsequent transmissions considering additional information from the receiver such as received data packets, duplicate confirmation and the measured PDR,

As we stated in [P1], we propose the RTO calculation by multiplying the estimated SRTT by a K parameter:

$$RTO = SRTT \times K \tag{5.1}$$

The SRTT calculation is the one proposed in RFC 6298 and is performed as follows: the first time an RTT measurement is obtained at time j, $SRTT_j = RTT_J$. The following calculations are in the following way, where SRTT is:

$$SRTT_{J+1} = (1 - \alpha)SRTT_J + \alpha(RTT_{(J+1)}) \quad (5.2)$$

where $\alpha = 1/8$, we take initially this value based on the RTO computation referred on RFC 6298 where it is calculating the SRTT using an EWMA (Exponential Weighted Moving Average). It gives more relevance to averaged samples of network condition than a recent network sample.

On the other hand, the RTT parameter is calculated as the time a publication packet is generated and its confirmation packet is received at the application level. The adaptation of the K parameter is necessary because this parameter should reflect the conditions of the network at a given time to estimate the value of RTO to appropriately react to lost packets or unexpected delay of packets.

Furthermore, we propose a verification window mechanism aimed at adapting the K parameter to be suited to the causes of the obtained PDR. The verification window is established whenever the RTT measurement is obtained, that is, every time a node sends a new publication packet (broker node or publisher node) until it receives an acknowledgment packet. Each time the verification window is established, we continuously calculate the PDR.

The PDR is dynamically calculated for each verification window based on information obtained from the destination node (broker node or subscriber node) related to the number of not duplicated packets received at the destination node. In addition, the node sending publication packets (publisher or broker node) keeps the number of packets sent (retransmissions included) for each verification window.

Once the PDR is calculated, we must infer the causes of the obtained PDR. The causes that we attempt to discover are:

CASE 1: There was no loss of publication packet.

CASE 2: Loss of publication packet (PUBLISH LOST).

CASE 3: Loss of Confirmation packet (PUBACK LOST).

CASE 4: Spurious retransmissions of publication packet (SPURIOUS RTX).

Once the cause of resulting PDR has been determined, the value of K may be adjusted depending on the situation. In this case, the new value for K parameter is:

$$K = K' \pm F \quad (5.3)$$

Where K' is the previous calculated value for K parameter and F, is the adjusting factor that increases or decreases the value of K according to the obtained PDR. For the adjusting factor of the K parameter, F, the used value is 0.5, for decreasing the value of K parameter (the case for lost packet) and a value of $F = 1$ is established for increasing the value of K parameter (the case for spurious retransmissions). The reason is that when there is data loss we must respond as soon as possible to recover the packet; thus, it is important to decrease slowly the value of K parameter to avoid "spurious" retransmissions. However, when there is "spurious" retransmissions the reaction must be more conservative to avoid duplicated packets on the subscriber node, thus we need to increment the value of K parameter rapidly. We evaluated several alternative F values, where, 0.5 and 1 presented the best results.

To be able to ascertain the causes of the obtained PDR, the RTO mechanism uses the information from the fields of the data packet (PUBLISH) and the confirmation packet (PUBACK). In this proposal we focused on MQTT-SN messages protocols used to send and confirm publication packets.

For the PUBLISH packet, it contains a field called Packet Identifier (PckId) which contains a unique identifier of that packet. In addition, we propose to add a field called "NumSeq" indicating the sequence number of the data packet being sent to the destination.

In the case of the PUBACK packet, the ReturnCode field indicates if the packet was received by the destination. We propose adding a "NumAck" field, which will contain the sequence number of the confirmation packet that is sent to the source through the packet PUBACK. Finally, we propose to create a new return code "0x02" for the "ReturnCode" field, which will indicate that it is not the first time that the confirmation packet is sent to the source. Therefore, in this situation the source could infer a PUBLISH duplicated packet caused by a confirmation packet lost.

Figure 50 shows the packet exchange where m indicates the Packet Identifier (PckId), s indicates the Sequence Number of PUBLISH packet and r represents the ReturnCode for PUBACK Packet. The value of "NumSeq" field of PUBLISH packet will be equal to one (1) for the first time this packet is sent to the destination. Besides the "ReturnCode" field will contain the value 0x01, which indicates that it has received and processed the corresponding PUBLISH packet by the destination.

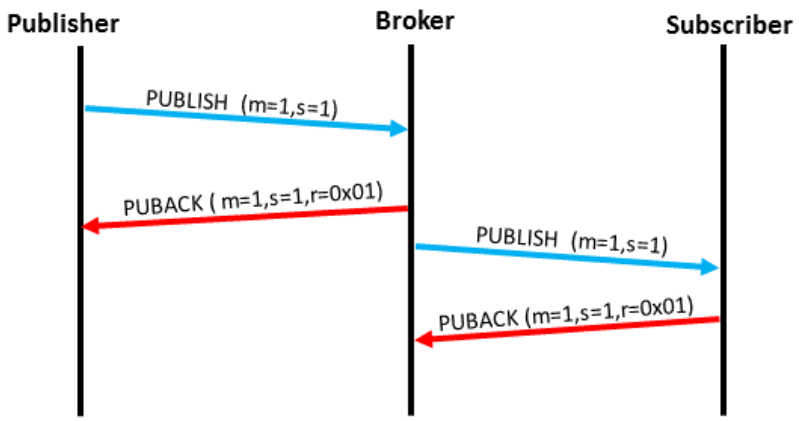


Figure 50. Packet exchange between nodes

In case there is a packet retransmission, the "PckId" field maintains the same identifier value, but the value of "NumSeq" increases sequentially, which is illustrated in Figure 51.

For PUBACK packet, the "PckId" field corresponds to the same value of the "PckId" field from the received PUBLISH packet. The "NumAck" field relates to the value of the "NumSeq" field from the received PUBLISH packet. In case the "ReturnCode" field value is 0x02, it indicates a PUBLISH duplicated packet as we explained above.

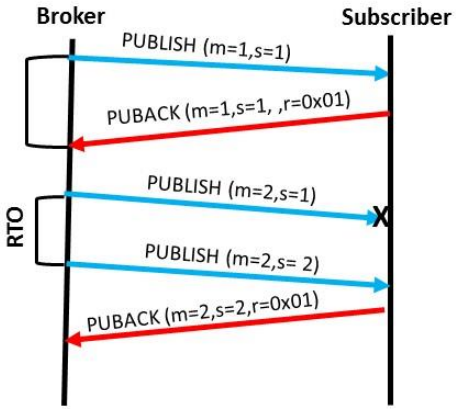


Figure 51. Packets retransmission

Taking into account the information obtained from PUBACK and PUBLISH packets we can infer the cases such as showed in algorithm 1 and Figure 52.

IF PDR == 1 THEN
No DATA LOST *(CASE 1)*
ELSE
IF (PUBACK Returncode == 0x02) then
PUBACK LOST *(CASE 3)*
ELSE
IF (PUBACK ReturnCode == 0x01) then
IF (Received PUBACK before the RTO expiration) then
SPURIOUS RETRANSMISSIONS *(CASE 4)*
ELSE
PUBLISH LOST *(CASE 2)*

Algorithm 1. Algorithm to infer the causes of the obtained PDR.

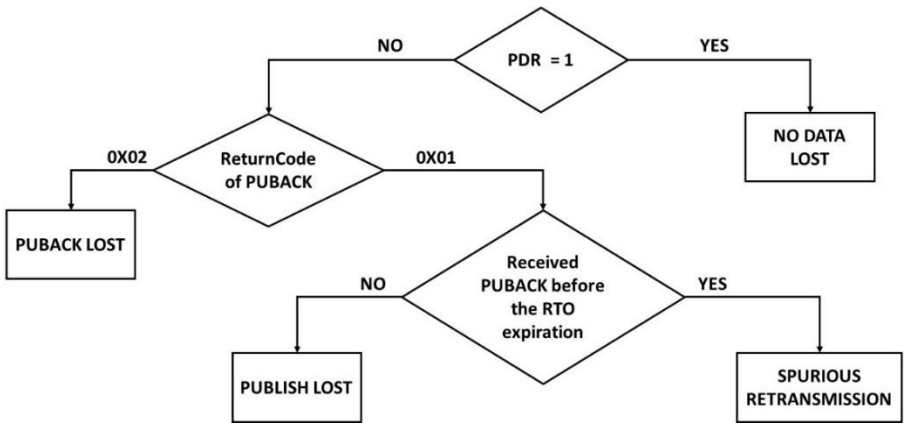


Figure 52. Flow Chart to decide the causes of the Obtained PDR

5.1.1.2 Computation of K Parameter

It is important to note as we mentioned before that publisher nodes generate Publication packets and the broker node receives them. The latter is responsible for sending publication packets to the appropriate subscriber nodes. Because of this way of working, the algorithm presented for calculating the RTO runs on both publisher nodes and the broker node, including the calculation of the PDR. The proposed algorithm uses an initially conservative value for K parameter equal to 4 and then it is subsequently adjusted according to network conditions.

Based on the following criteria we choose the initial value of K: it is necessary an initial K value to avoid spurious retransmissions but not too large to avoid long time reaction to lost. Several alternative K values for small and large RTO values were evaluated, where, 4 presented the best results.

Figure 53 shows how the K parameter is adapted for case 1, case 2, case 3 and case 4 above mentioned. In this figure, m indicates the PckId and *i* represents the Sequence Number of PUBLISH packet.

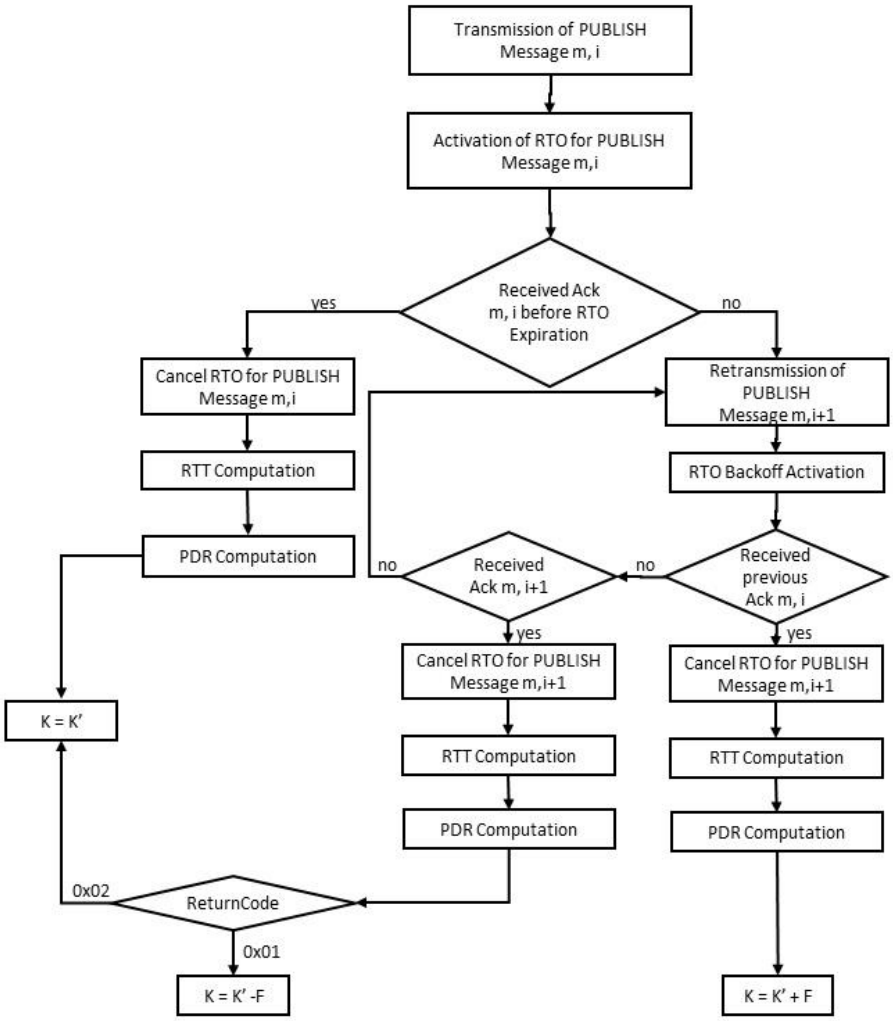


Figure 53. Algorithm for adjusting the K parameter

It is important to note in this chart that PDR computation value is equal to 1 when it receives an ACK before RTO expiration. Moreover, PDR is less than 1 when it receives an ACK after at least the first retransmission. We have represented this process in the chart only to clarify the flow of the algorithm for the computation of the K parameter.

In addition, the technique of deriving the value of K parameter could be summarized in equation (5.4).

$$K = \begin{cases} K' + F, if\ Received\ Previous\ Ack\ m, i\ after\ a\ RTO \\ K' - F, if\ (Received\ Previous\ Ack\ m, i+1\ before\ a\ RTO)\ and\ (ReturnCode = 0x01) \\ K', otherwise \end{cases} \quad (5.4)$$

Figure 54 depicts an example for each case to calculate the Verification Window previously detailed. In this figure, m indicates the PckId and i indicates the Sequence Number of PUBLISH packet also r represents the ReturnCode for PUBACK Packet.

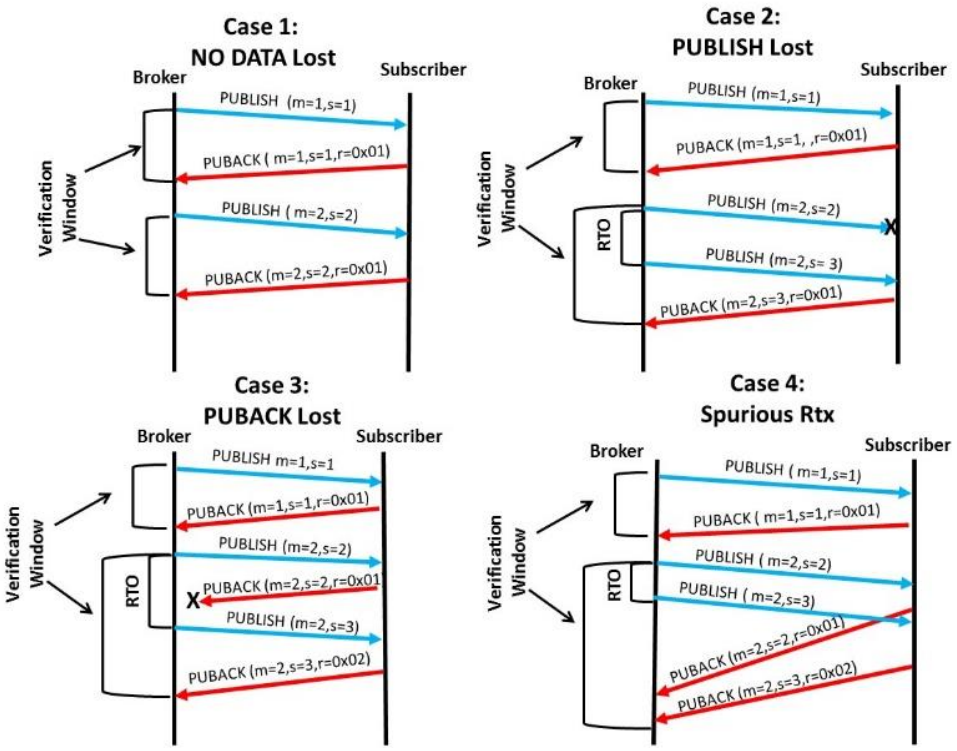


Figure 54. Verification Window calculation for each case.

The new approach we propose to deliver a reliable packet in WSN using Publish/Subscribe model is in accordance with some aspects of the CoAP standard protocol and CoCoA mechanism. For example, the used publication discipline is of great importance related to the PDR performance metric evaluation. We adopt the CoAP publication discipline. It consists on giving priority to sending the new publications rather than attempting to retransmit the old ones. In this case, the PDR could be decreased because a packet lost will not be recovered when the publisher or broker node generate a new publication packet.

5.1.2 QoS Level 2 (Data Aggregation)

This QoS level adds to the new RTO mechanism the data aggregation capability. In most of WSN applications, sensor nodes deployed in common area, could transmit similar or same data that could result in redundant data in the network. That results on consumption of valuable resources and in network congestion. Depending on application requirements, aggregated sensor data is enough. By using data aggregation, several data values (messages) transported though the same path can be aggregated to reduce the amount of traffic and thus reducing the network congestion and helping to preserve scarce resources such as bandwidth and energy consumption. In our approach, the node playing the broker role will perform data aggregation of the next two messages being received after a specific case has occurred (publication lost, a publication acknowledgment lost or a spurious retransmission), and then transmits this in a single packet to the subscriber node. For data aggregation, we have considered two or more messages. The evaluations carried out in this thesis, only show the results in which we consider two messages since the number of messages delivered could decrease as more messages we aggregate. This is due to the CoAP publication discipline we use, and when we have more than two aggregated messages there is a higher probability that the data is lost.

Our proposal considers the aggregation of messages only between broker node and subscriber node that requests QoS level 2. It is not realized from publisher node to broker node because other subscriber nodes have different QoS requirements, so data aggregations could be not suitable for them.

This single packet contains each data value corresponding to the two previous packets. For example, the first packet contains the temperature data value 33. A second packet is received containing the data value 40. In this case, the broker node puts these values in a new packet, which is transmitted to the subscriber node. This way to aggregates data is called Lossless Data Aggregation [71] and it provides a way to send data values that are considered important for the application.

We consider this QoS level is oriented to messages that are not critical in the “arrival on time” target to the destination node, so that there is no conflict with real time sensor readings. It is important to note that we do not propose a new data aggregation method as a novelty. We use a data aggregation mechanism to avoid an increase of network congestion if a node continues transmitting in a congested medium, by reducing energy consumption and the use of network bandwidth, which are other important performance metrics on WSN.

5.1.3 QoS Level 3 (Timeliness)

Before detailing our proposal to provide timeliness mechanism based on deadline, it is important to mention that previously a contribution was made [P6] to add timeliness to the CoAP protocol through a priority mechanism in the delivery of messages according to its category. From the contributions made to this work was derived the need for a mechanism that not only sends based on priority but considers an established deadline. In this section we will present a brief summary of the first proposal and later we will detail the deadline mechanism.

5.1.3.1 Priority support for Observe model of CoAP protocol

The proposed mechanism allows to the subject in the observe model [21] of CoAP protocol selects the order which send updates of events (publications) based on two categories: critical and non-critical. In addition, the mechanism provides to each observer a way to indicate the update categories they want to receive. In this manner, each observer would only receive the updates based on its role in the application and the subject would avoid sending updates to all nodes in the network which would result in the decrease of the experimented delay and bandwidth and energy saving.

The proposed mechanism provides four QoS levels for updates delivery priority and categories: low, medium, high and highest. The two most significant bit of the observe option value are used to specify the value of the QoS level that we call QoS field. The different values defined for QoS levels are the following:

- QoS Level 1: it is represented with the value 00 in the QoS field and it indicates that the subject will send non-critical and critical updates with low priority.
- QoS Level 2: it is represented with the value 01 in the QoS field and it indicates that the subject will send both non-critical and critical updates with medium priority.
- QoS Level 3: it is represented with the value 10 in the QoS field and it indicates that the subject will send only critical notification with high priority.
- QoS Level 4: it is represented with the value 11 in the QoS field and it indicate that the subject will notify with the highest priority only the start and the end of a critical state.

5.1.3.2 Deadline mechanism

As we previously explained, the full time and space decoupling and scalability of the publish/subscribe model, makes the broker node the key point of decision to deliver the packets to subscriber nodes with different QoS levels, for example based on a deadline target previously specified by subscriber node. In contrast with [66], where each node takes the deadline decision, they do not provide the decoupling we previously mentioned. Moreover, it is very complex to manage different QoS levels because each subscriber node would need to know the offered QoS level for each publisher node and publisher nodes would need to maintain a list of subscription and requested QoS level for each subscriber node. This way to work is not scalable and the resources on memory and processing resources could be high for the node.

We propose a deadline mechanism (DM) that is explained as follows. A Smooth Delay calculation for each received packet from publisher nodes to broker node (SDP2B) for instant j+1 is calculated by:

$$SDP2B_{J+1} = (1 - \alpha)SDP2B_J + \alpha(DP2B_{J+1)} \quad (5.5)$$

where, DP2B j+1 is the measured delay for each packet from publisher node to broker node, α = 1/8, and SDP2Bj is the SDP2B for instant j. As we mentioned above, the α value is based on the RTO computation referred on RFC 6298 where it is calculating the SRTT using an EWMA (Exponential Weighted Moving Average).

In the same way showed in equation (5.6), we calculate a Smoot Delay for each received packet from broker to subscriber node (SDB2S) using:

$$SDB2S_{J+1} = (1 - \alpha)SDB2S_J + \alpha(DB2S_{J+1}) \quad (5.6)$$

where, DB2S j+1 is the measured delay for each packet from broker node to subscriber node, α = 1/8, and SDB2Sj is the SDB2S for instant j.

Then we use the SDP2B and SDB2S values to calculate the Deadline Factor (DF), jointly with the Certainty level (CL). The CL is the probability of delivering the packet to the subscriber node based on the previous measured delay from broker to subscriber node. Then, the DF that controls the deadline mechanism is expressed as showed in equation (5.7).

$$DF = (SDP2B + SDB2S) \times CL \quad (5.7)$$

Finally, the DF value is compared with the Deadline Target (DT) that was previously established by the subscriber node. A packet with DF < DT is transmitted to the subscriber node. Otherwise, it is discarded.

We consider that a deadline mechanism at application level that considers network conditions is vital to ensure a timeliness for delivering packet. In conclusion, it is important to point out that this QoS level does not aim for the subscriber node to obtain a better PDR, but the packets arrive to the subscriber node as determined by the target deadline. To cope with that, the broker node selects the packets that have a higher probability to achieve the subscriber node in accordance with equation (5.7). In this equation, the delay is calculated as a measurement of the network condition. Therefore, those packets with a high requirement of timeliness would have higher probability to be discarded depending on the network conditions.

5.2 Experiment Setup for Priority support for Observe model of CoAP

In this section, we present the description of testbed in a real WSN and the performance metrics to evaluate the proposed mechanism that provides priority support for observe model of CoAP.

5.2.1 Experiment Environment

We consider a scenario where WSN has the goal of monitoring cardiac rate of a patient. The proposed mechanism is suitable to this environment because this kind of e-health[93] application has strict deadline. The WSN consist of 6 observers which are interested on receive update of events about the state of a patient. Each of them has different priority and information of interest such as showed in the Table 5.

Observer	Information	QoS Level
Doctor	When the patient enters or leaves a critical state	4
Nurse	Each state of the patient	2
Alarm	When a critical event starts or it ends	4
Personal Monitor	Each state of the patient	2
General Monitor	Each state of many patients	1
Intra Venous (I. V.)	Each critical information	3

Table 5. QoS level for each observer in the e-health scenario.

The deadline of the updates corresponds to the sampling rate of the cardiac rate. For critical updates, the deadline is 100 ms and sending a 2 bytes of data payload. In the case of non-critical updates, it performs data aggregations of 37 samples in a single packet, therefore the deadline for this information is 3700 ms and sending 74 bytes of data payload in an 802.15.4 frame.

We consider a star topology, as showed in Figure 55 where nodes are implemented on TelosB platform. In addition, the observers are equally spaced between each other and at the same distance from the subject to keep simplicity

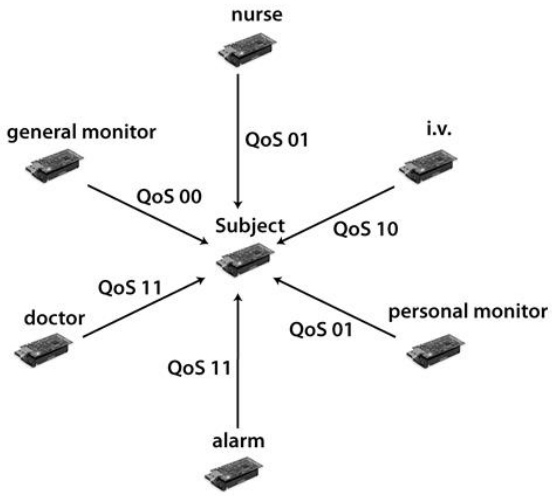


Figure 55. Topology for the WSN testbed

5.2.2 Performance metrics

We consider latency, delivery ratio and energy consumption as metrics to evaluate the proposed mechanism.

- **Latency**: it refers to the time interval from the update is created in the subject until the observer acknowledges its reception.

- **Delivery ratio**: it defines the probability to receive an update by the observer.

- **Energy consumption**: it refers to the total consumption of energy based on sending and receiving updates or acknowledgments. It does not consider the energy consumed to listen the channel. We use the device Agilent Technologies DC power Analyzer N67705A to obtain the energy. The energy is sampled each 0.02 ms.

5.3 Simulation Setup for QoS level mechanism

In this section, first we present the description of the simulation environment and the used performance metrics, to evaluate each proposed QoS Level. Finally, we analyse and discuss the obtained results from the evaluation of each QoS level by means of simulations.

To evaluate that our proposal is better than other protocols using fixed and adaptive RTO, we compare the PDR values for subscriber with QoS level 1. In this last case, we show the impact of data aggregation to reduce the network congestion and increase the probability of message delivery. Finally, we present the evaluation of timeliness for publication packets with the objective of delivering packets to the subscriber nodes on time.

5.3.1 Simulation Environment

Simulation experiments were carried out using OMNet++. We consider a local WSN where the goal of the application is the monitoring and control of critical environmental parameters such as temperature and humidity in electrical substation[95] through a WSN deployment. In this context, two types of devices are deployed in different parts of the application area: publisher and subscriber nodes. The publisher nodes are responsible for measuring the critical parameters in the electrical substation. Additionally, there are four subscriber nodes (each for a different QoS level). In addition, each publisher node sends a packet in average each 5 seconds and in total, each publisher node sends about 100 packets to the broker node, and this one to the corresponding subscriber node. We consider a multi-hop scenario consisting on nodes that are located up to 3 hops from the broker node and at equal distance among them, which is illustrated in Figure 56.

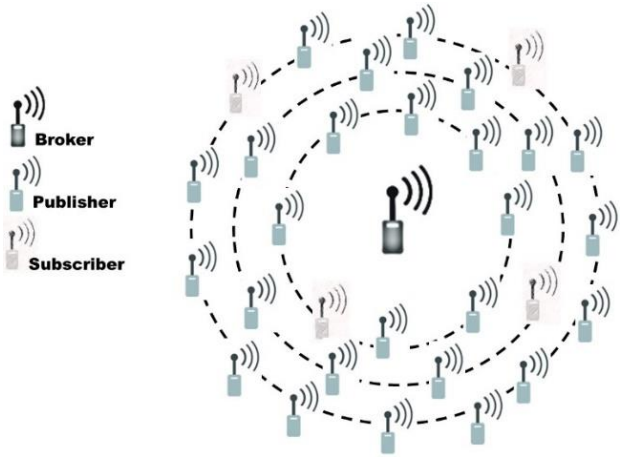


Figure 56. Simulated network topology

Packets originated at the publisher node are routed to the broker node through multiple relay nodes. When these packets arrive to the broker node, the application layer of this node generates a new publication packet with the desired QoS level for the corresponding subscriber node. This new publication packet is routed in a similar way to the subscriber nodes as shown in Figures 57 and 58. Relay nodes could be also publisher or subscriber nodes. Also, in these figures it can be noted the coverage area for each node which allows the network communication.

The simulation scenario consists of a number of publisher nodes (10, 20, 30, 40 and 50). We fixed 4 subscriber nodes. For this work, we have considered four subscriber nodes, each one with a different QoS level. This is because we are focusing on an application scenario where the same publication message sent by the publisher nodes is required by each

subscriber node with a different QoS level. However, a QoS level 1 is always considered between publisher and broker nodes to guarantee a reliable communication in this way.

For example, the first subscriber does not require reliability on packet delivery (best effort), the second one requires reliability on packet delivery, the third one requests reliability and could accept data aggregation if it is necessary, and finally the fourth one is interested in timeliness of a packets. We place Subscriber nodes so that we can study the most pessimistic scenario of each QoS level. In this work, we use static routes among nodes to eliminate the delay caused by the implementation of routing protocol for routing decision. The implementation of a routing protocol and its effects is out of the scope of this study and would be considered for future work. Each simulation experiment lasts for 500 s. Each point in the graph presented in this section is the average of 10 simulation runs.

We use the 2.4 GHz range with a bandwidth of 250 kbps based on IEEE 802.15.4 [31] for the PHY layer, with a maximum number of MAC-layer retransmissions set to 3, which is the default value of IEEE 802.15.4 [31]. We use the current consumption parameters based on Zolertia Z1 [94] datasheet to obtain the results for energy consumption of the nodes.

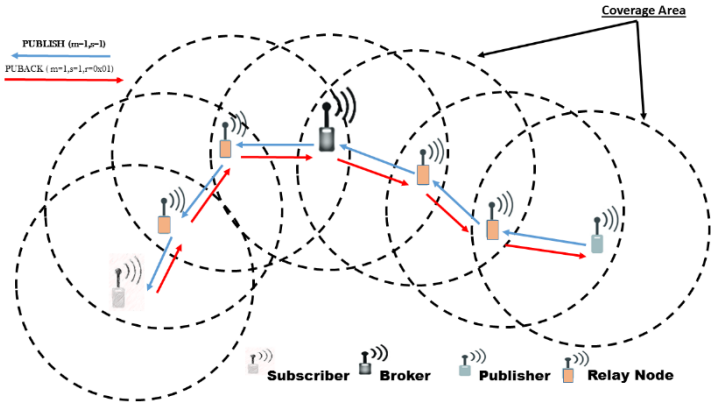


Figure 57. Flow of packet with Subscriber node at one hop from Broker node.

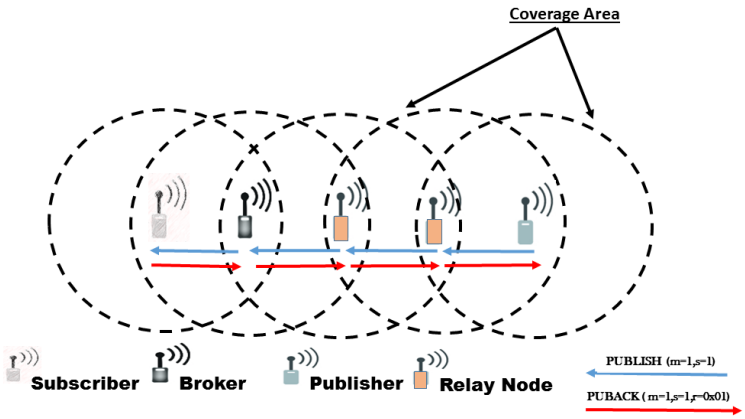


Figure 58. Flow of packet with Subscriber node at three hops from Broker node.

5.3.2 Performance Metrics

In this section, we define the used performance metrics.

- Packet Delivery Ratio (PDR): It expresses the total number of publication packets received by each subscriber node divided by the total number of publication packets generated by all publisher nodes of the events to which the subscriber node has been subscribed. It does not take into account duplicated publication packets received by subscriber nodes.

- Message Delivery Ratio (MDR): It expresses the total number of publication messages received by each subscriber node divided by the total number of publication messages generated by all publisher nodes of the events to which the subscriber node has been subscribed. It does not take into account duplicated publication messages received by subscriber nodes.

- Packet Timeliness Ratio (PTR): it expresses the number of delivered packets before deadline divided by the total number of publication packets generated by all publisher nodes of the events to which the subscriber node has been subscribed. It does not take into account duplicated publication packets received by subscriber nodes.

- RTX Packet Ratio (RPTXR): It is the ratio of the total number of publication packets retransmitted, divided by the total number of sent publications packets. This metric is evaluated for the total number of publisher nodes.

- RTX Message Ratio (RMTXR): It is the ratio of the total number of publication messages retransmitted divided by the total number of sent publications messages. This metric is evaluated for the total number of publisher nodes.

- Duplication Packet Ratio (DPR): This indicates the ratio of the number of duplicated publication packets received to the total number of publication packets received. We evaluate this metric in the subscriber node.

- Duplication Message Ratio (DMR): This indicates the ratio of the number of duplicated publication messages received to the total number of publication messages received. We evaluate this metric in the broker node.
  g
- Energy consumption: this metrics refers to the total amount of consumed energy (J) by the broker node to transmit the total number of publication packet for each subscriber node with the corresponding QoS level. We evaluate this metric in the broker node.

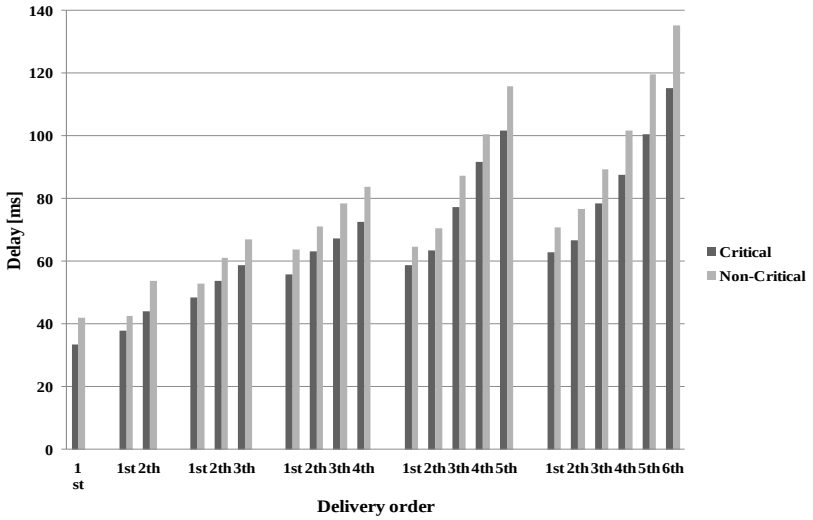
5.4 Results and Discussion for Priority support for Observe model of CoAP

5.4.1 Latency

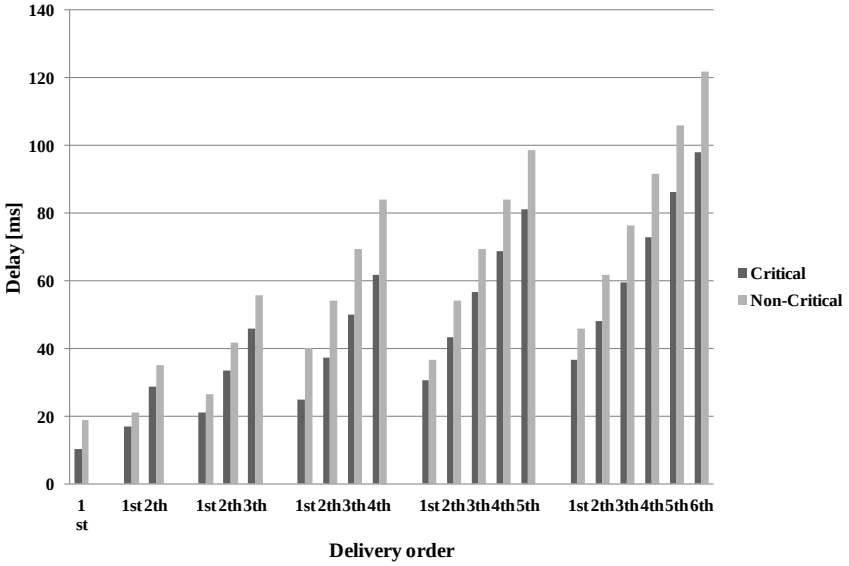
The results in Figure 59, show the delay of updates as a function of the delivery order and each value represents the average delay of 100 delivered updates. The experienced delay by observer with non-critical updates is significantly lower than update deadline.

In addition, our mechanism achieves a delay reduction because it only delivers updates to observers requiring non-critical updates. In this case, there are only three nodes (observers) requiring non-critical updates from the total of six observers in our network scenario, as shown in Figure 56

On the other hand, observers with critical updates meet the update deadline requirements because our mechanism delivers only required updates depending on specified QoS level by observer as shown in Figure 59, which results in lower experienced delay, lower energy consumption and network bandwidth. In our scenario there are only four nodes that receive critical updates. The current delivery mechanism of observer option could not able to fulfil this requirement in emergency applications where critical updates are an important paramount.



(**a**)



(b)

Figure 59 Delay as a function of the delivery order. A) Persistence B) Best Effort

We also evaluate the reliability support of observe option. In this situation, as we expected there is a growth of the delay because of the persistence mode of observe option as shown in Figure 59. In this case, there is a contention for accessing the channel between the subject sending the update and observer sending the CoAP ACK which results in one of them will perform several attempts for transmitting the packets if the channel is busy. This situation and the increase of number of nodes leads to a higher delay. Also, we have to consider the publication discipline of CoAP which cancels any update retransmissions if a new one is generated. Therefore, the extra delay caused by collision can be either due to a loss of data or due to the effect of publication discipline of CoAP.

5.4.2 Delivery Ratio

We evaluate the delivery ratio as a function of the delivery order as shown in Figure 60 comparing persistence and best effort mode. In persistence mode, we consider an RTO value of 38 ms which is equivalent to the sum of the average delay experienced in presence of a single observer and its standard deviation. The retransmission timer does not consider the time when the update is generated. It is only started when the update is sent.

As one can note, the observers receiving updates with high or highest QoS level get a high delivery ratio. In persistence mode, there is a less channel contention because observers only receive critical updates which results in less chance to collide with an ACK. In addition, the observers have more time if there was a need of update retransmission before CoAP publication discipline is activated. In best effort mode, observer get a higher delivery ratio compared with persistence mode because the subject does not have to compete with anyone which results in that probability of collision is inconsequential. These results only apply for a WSN star topology. In multi-hop network we would expect that the persistence will achieve better results, but the study of this topology is out of the scope of this evaluation.

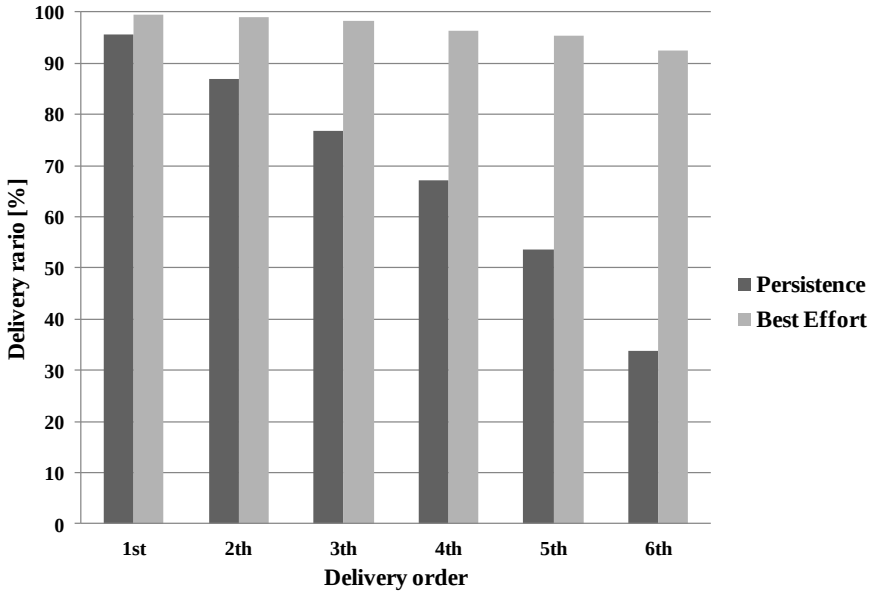


Figure 60. Delivery ratio as a function of the delivery order.

5.4.3 Energy Consumption

In this section we test the energy consumed by subject to send critical and non-critical notifications in best effort and persistence mode. As shown in Figure 61, the persistence mode implies a higher energy consumption compared with best effort. This is because there is a higher number of messages in persistence mode caused by ACK messages from observers. The subject has to receive and to process any ACK messages which results in that radio chip waste extra energy. In contrast, in best effort mode, as one may expect, the subject consumes less energy because there are not additional messages.

As we explained previously, our proposal allows an observer to choose the updates to receive based on QoS levels. The results showed that this ability reduces the energy consumption of the subject. The subject sending non-critical updates consumes less energy. For critical updates, the energy consumption is also considerably reduced because subject is sending only the updates to those observers that selected the QoS level 4.

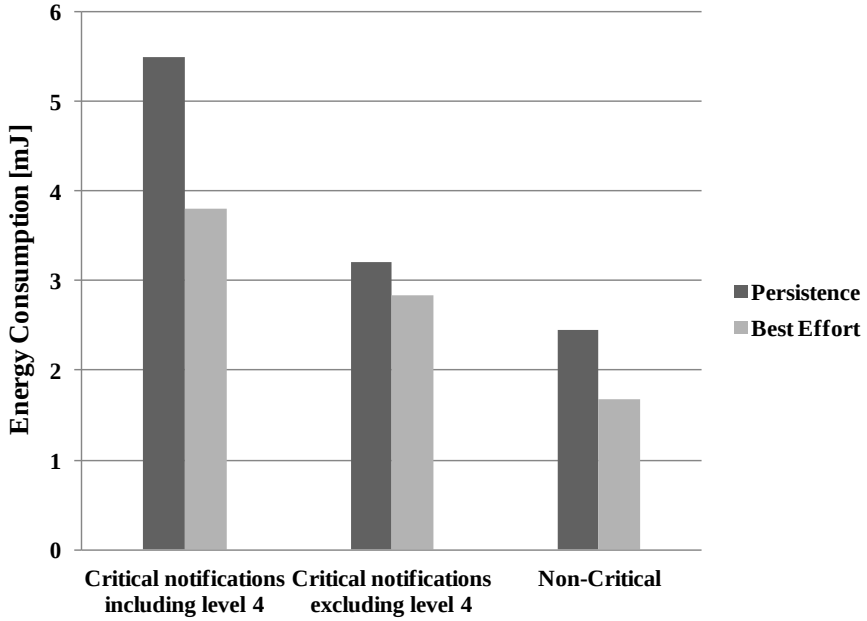


Figure 61 Figure 61 Energy consumption of the subject.

5.5 Results and Discussion for QoS level mechanism

5.5.1 Packet Delivery Ratio (PDR): QoS level 1

As we mentioned above in Table 2.1, the most of proposal providing reliability are based on fixed retransmission scheme and in general, they are very similar with MQTT-SN and CoAP. Thus, we have chosen these protocols to compare it with our proposed dynamic method for RTO calculation. On the other hand, CoCoA proposes a mechanism to adapt RTO using network conditions in a similar manner as our proposal; therefore, we also compare it with our proposal.

We observe that, the highest PDR for the subscriber node with QoS level 1 is obtained when using our adaptive RTO method. As illustrated in Figure 62, subscriber node increases

the PDR up to 43% compared with the MQTT-SN protocol and up to 26% compared with the CoAP protocol. On the other hand, in the most cases, the highest PDR for the same subscriber in most of the cases is obtained when using K Parameter compared with CoCoA. In this situation, the subscriber node increases the PDR up to 3% compared with CoCoA.

One of the reasons for that is that our proposal adjusts a K parameter depending on the calculated PDR on each verification window for the subscriber node. That is, our proposal uses sequence numbers and acknowledgment numbers that allow infer a data loss, an acknowledgment loss, or a spurious retransmission and act in consequence. In contrast, MQTT-SN defines a fixed RTO value of 10 to 15 seconds, and CoAP protocol chooses an RTO value from an RTO interval. Both of them do not consider any network conditions to adjust the RTO. This situation results in a low capacity to reaction to data, acknowledgment loss or a spurious retransmission and thus the subscriber node receives a lower PDR. Furthermore, CoCoA considers only RTT as a network conditions parameter, although this is good criteria, but we consider it is not enough to adjust RTO.

On the other hand, an important aspect to take into account in the PDR is the effect caused by the publication discipline. We adopt the CoAP publication discipline. It consists on giving priority to sending the new publications rather than attempting to retransmit the old ones. Considering this situation, it can be noticed that using fixed value for parameter K could result in greater RTO value which results in retransmission activated too late to recover publication packets in the case of loss.

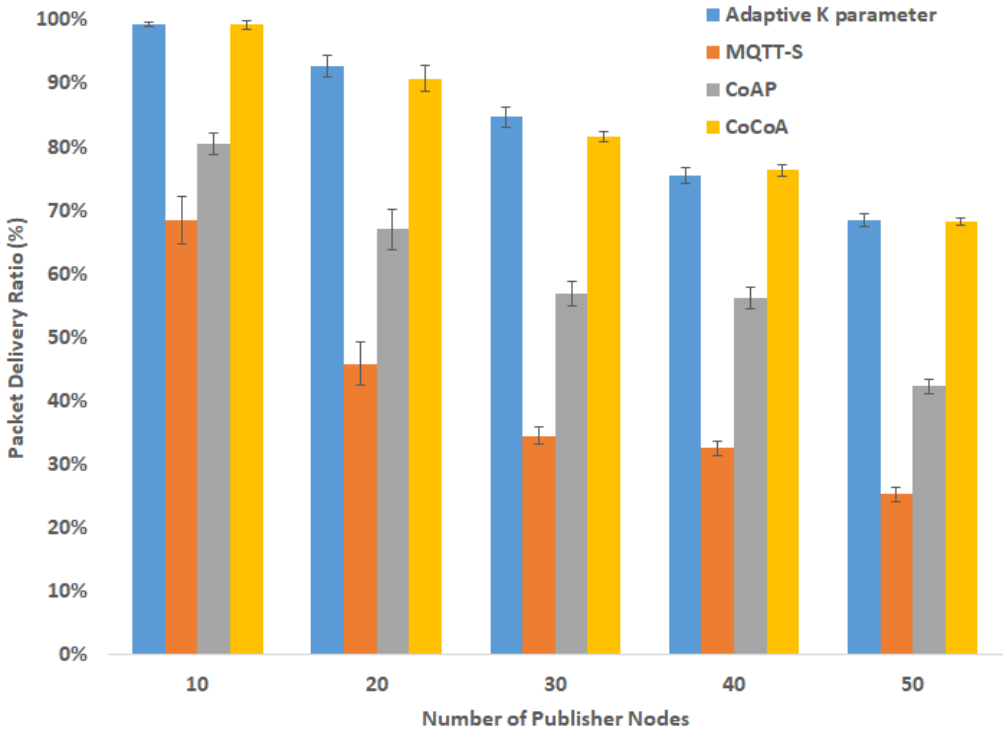


Figure 62. PDR Comparison for QoS Level 1 using our proposal with other protocols

As we have observed in this section, the subscriber node using our adaptive RTO method gets a higher PDR than the other shown protocols.

5.5.2 Message Delivery Ratio: QoS level 2

In the presented results, the broker node aggregates into single packet the collected values from the next two packets after a publication packet lost, publication acknowledgment lost or spurious retransmission has occurred previously for the subscriber node. We evaluate the effect of the Data Aggregation by evaluating the PDR at the Subscriber node taking into account the same effect in the Subscriber Message Delivery Ratio (MDR). As we expected, results showed that PDR for subscriber node with QoS Level 2 was up to 37% lower than subscriber node with QoS Level 1, as can be seen in Figure 63. One of the reasons for this is that the broker node stops the packet transmission to subscriber nodes with QoS Level 2 each time the data aggregations is performed. At this moment, channel occupation is reduced thus the other subscriber nodes take advantage to receive publication packets and to access the channel with less contention to send its confirmation packets.

In contrast, one may observe in Figure 64, that the MDR for subscriber node with QoS Level 2 was up to 18% greater than the one obtained by subscriber node with QoS Level 1. The reason of this is that each received publication packet by subscriber node with QoS level 2 could contain up to 2 messages, thus it increases the amount of received messages in comparison with the others subscriber nodes that only could receive one message for each received packet. It can be noted that the network congestion increases as the number of publisher nodes increases, which in turn results in an increase of packet losses. In this case, the subscriber node with QoS Level 1 will attempt to recover each lost packet by retransmitting it (1 message). However, this situation could lead to increase the network congestion and thus the subscriber node with QoS Level 1 gets a lower MDR. On the other hand, the same situation could happen for the subscriber node with QoS Level 2, but in this case, data aggregation helps recovering a higher number of publication messages, because of each retransmitted publication packet could contain up to 2 publication messages.

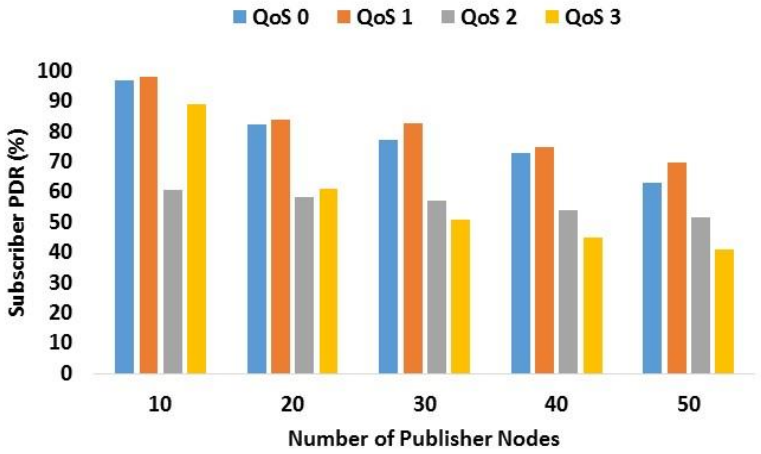


Figure 63. Subscriber PDR Comparison between different QoS Levels

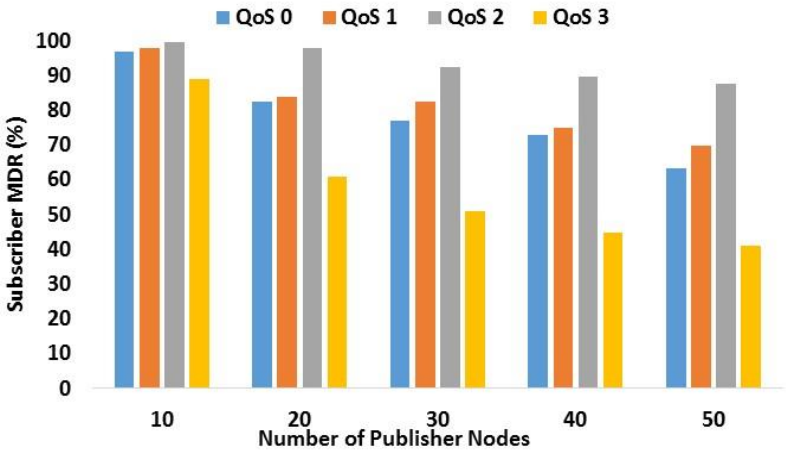


Figure 64. Subscriber MDR comparison between different QoS Levels

Although data aggregation has been added, the subscriber node does not decrease its number of received messages regarding messages generated; it demonstrates the feasibility of our proposal for this kind of networks.

The purpose of both figures is to demonstrate the differences between QoS level 1 and 2. QoS level 2 carries out Data aggregation, which results in an increase of the number of messages received by the subscriber node. This is because two data messages are put into a packet each time data aggregation is activated. Figure 63 shows that subscriber node with QoS level 2 gets a lower PDR than QoS level 1. The reason is that PDR metric considers the received packets by the subscriber node instead of the received data messages that considers the MDR metric. In contrast, if we consider the received messages by subscriber node as shown in Figure 64, it is clear that subscriber node gets a number of received messages greater than subscriber node with QoS level 1. Finally, the choice of the QoS level 1 or 2 depends on application requirements. The use of QoS level 2 could increase the delay due to the time required to perform data aggregation. We recommend the use of QoS level 1 in the cases that application requires the packet of delivery with reliability once the publisher nodes have generated them. On the other hand, if the application requires reliability but it is able to tolerate delay in the delivery of packet we recommend QoS level two. Finally, if requirement of the application is to receive data packets with a deadline target, the QoS level 3 is suitable because it provides timeliness.

5.5.3 Retransmitted Packet Ratio and Retransmitted Message ratio

In Figure 65(a) one may observe that retransmitted packet ratio is up to 4% (50 nodes) lower for subscriber node with QoS level 2 than subscriber node with QoS Level 1, the reason of that is because of the data aggregation process which allow retransmit up to 2 publication messages into a single publication packet. In contrast, Figure 65(b) shows that the subscriber node with QoS level 2 requires up to 8% higher publication message retransmitted than subscriber node with QoS Level 1. Although the number of retransmitted messages for subscriber node with QoS Level 2 is higher than subscriber node with QoS Level 1, as illustrated in Figure 65(b), this number of retransmissions is necessary to recover the publication message to obtain the highest PDR.

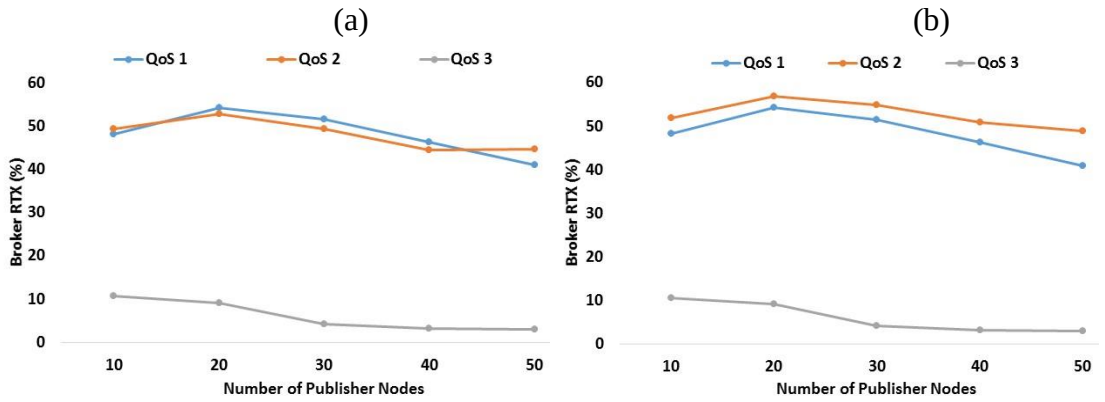


Figure 65. Effect of the number of Publisher nodes on the Broker RPTXR (a) and Broker RMTXR (b) using different QoS Levels

5.5.4 Duplicated Packet and Duplicated Message Ratio

In the case of Publication Duplicated Packet Ratio (DPR), in Figure 66 the subscriber node with QoS Level 2 gets in both cases up to 9% (50 nodes) greater number of duplicated publication messages than subscriber node with QoS Level 1. That is due to, among other reasons, publication acknowledgment lost which in turn results in retransmission of publication message. In this situation, each received duplicated publication packet by subscriber node with QoS Level 2 could contain up to 2 publication messages thus increasing the number of duplicated publication messages.

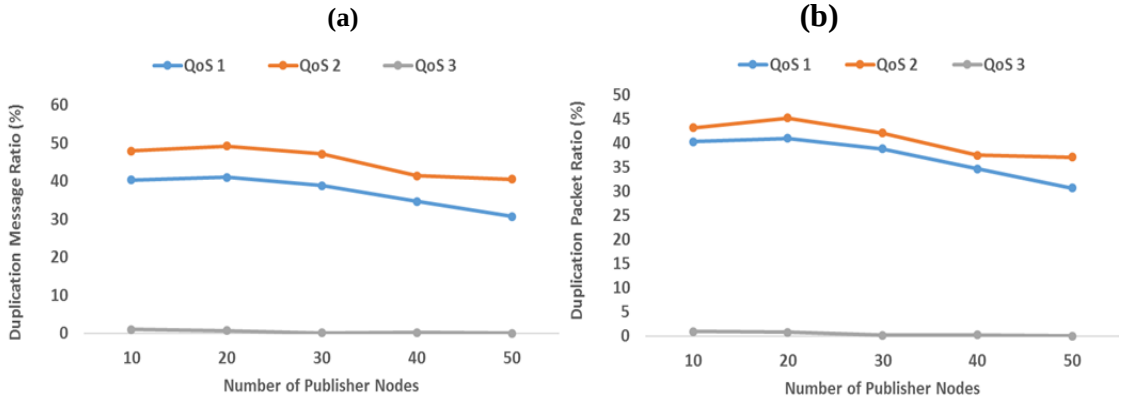


Figure 66. Effect of the number of Publisher nodes on the Duplicated Publication Message Ratio (a) and Duplicated Publication Packet Ratio (b) from Subscriber Node using different QoS Levels

5.5.5 Packet Timeliness Ratio: QoS Level 3

We consider a useless packet if a packet has not been delivered before the deadline expires. In this section, we evaluate the effect of timeliness on subscriber node using our proposed QoS level 3. As we can observe in Figure 67, QoS level 3 ensures the timeliness

delivery of packets up to 81%. It is also important to note that although the network load increases, the decrease of the packet timeliness ratio is reduced. It is important to emphasize that this QoS Level is focused on timeliness of delivered packets to the subscriber node. Broker node is in charge of transmitting only those packets that the deadline mechanism considers that they are able to accomplish with the deadline target. We are not interested on subscriber PDR obtained, because it could be lower as more packets are discarded.

In contrast, we could deduce that the energy consumption is lower using QoS level 3 because broker node would discard every packet not accomplish with the deadline target.

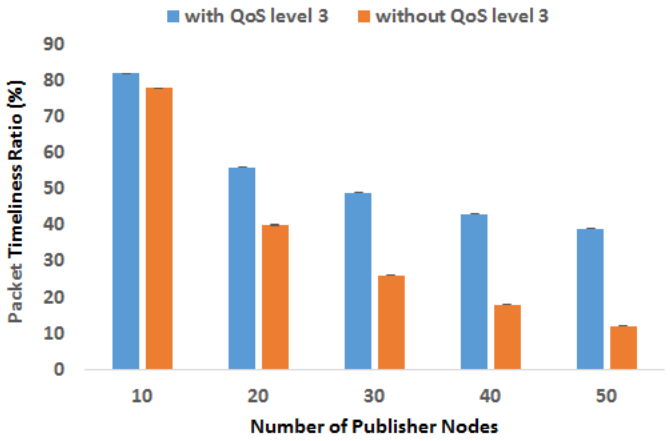


Figure 67. Effect of using Timeliness (QoS Level 3) for packet delivery

5.5.6 Energy Consumption

We focus on the energy drawn by broker node which is responsible to transmit publication packets to each subscriber node with corresponding QoS level. As we can observe on Figure 68 the results show that publication packets transmitted with data aggregation (QoS level 2) consume lower energy than without this feature (QoS level 1). One reason of that is because broker node reduces the number of transmitted packets to Subscriber node.

In contrast, we get the lowest energy consumption with QoS level 3 and QoS level 0. That is because, this QoS level is not focused on reliable packet delivery. QoS Level 3 as we above mentioned is focused on timeliness. Therefore, the amount of energy drawn is proportional to the amount of publication packets transmitted before deadline expire. On the other hand, publication packets are transmitted on best effort for QoS level 0. Therefore, there is not addition of energy consumption for retransmission compared with the others QoS levels.

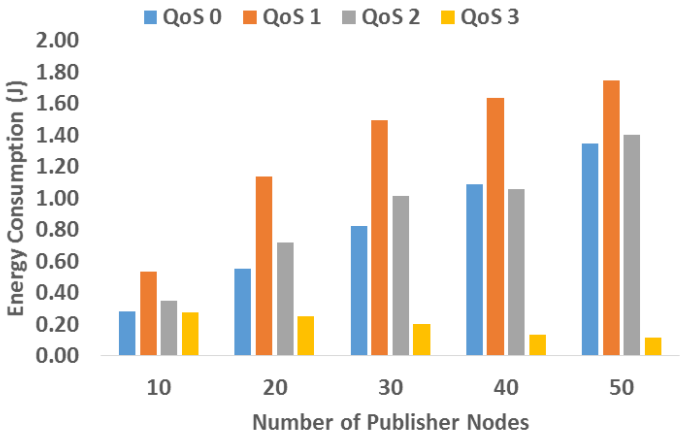


Figure 68. Energy Consumption of Publication packets for each QoS Level

5.6 Conclusions and Contributions

We have proposed a mechanism that enables to applications to express different QoS requirements through of three QoS levels. We focus mainly on reliable packet delivery with the option of data aggregation support and timeliness on WSN based on Publish/Subscribe model to meet applications requirements.

The QoS Level 1 provides a reliability in the delivery packet through an adaptive RTO mechanism that adjusts a K parameter value depending on the subscriber node PDR. The evaluation results showed that subscriber obtains an increase in PDR. The QoS Level 2 is similar to previous QoS Level 1 but in addition, it provides data aggregation support in order to reduce network congestion situation. We show with the results that QoS Level 2 provides a higher MDR on the subscriber node comparing with QoS Level 1. Furthermore, the presented results showed that the simple aggregation mechanism could help to reduce congested situations and to improve performance.

Finally, QoS Level 3 provides timeliness on delivery packet as we showed in the obtained results. Firstly, we presented the contribution of a novel mechanism to adding QoS support for timeliness to the observe CoAP option. In this case, we proposed a mechanism which allow to the observers to choose the level of priority to receive updates. We have defined four level of priority which establish the order and class of updates (critical and non-critical) that the observer will receive. The results demonstrate that the delivery order of an update has an important effect in the delay and delivery ratio experienced in the observer. Also, the definition of classes we have established reduces the energy consumption of the subject, and the probability of collision which results in lower delay and higher delivery ratio on observers. However, in this mechanism the subject is the only one in charge of identifying the class of an update and deciding the class and order of the updates to send to observers. Therefore, observers which could have different requirements would be not considered by the subject. This is a limitation of this approach which would result in the impossibility of meet their requirements. Based on this contribution we proposed a deadline mechanism to provide timeliness as a QoS level. In this mechanism, the Broker node is in charge of transmitting only those packets that the deadline mechanism considers that they are able to accomplish with the deadline target. The results showed that packets sent to subscriber nodes were received before the defined deadline using our proposal also we achieve a reduction of

energy consumption because broker node would discard every packet not accomplish with the deadline target. In this QoS level we are not concerned on subscriber PDR obtained, because it could be lower as more packets are discarded.

In conclusion, we consider our proposal provides QoS features support for applications which need meet different requirements. As mentioned above, for a smart way monitoring of critical infrastructures (such as electrical substation) it is important to take into account the reliable delivery packet and timeliness, which requires that the occurrence of an event is near real-time notified.

The results of the research presented in this chapter led to the publication of the paper *"Adding QoS support for timeliness to the observe extension of CoAP*" [P6] which was presented in 8th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). In addition, we published the article *"Publish/Subscribe Protocol in Wireless Sensor Networks: Improved Reliability and Timeliness"* in KSII Transactions on Internet and Information Systems [P2].

6. Conclusions and Future Works

Throughout the work carried out in this thesis we have proposed several mechanisms that allow the adequacy of the Publish / Subscribe protocols to the characteristics and limitations of the WSN for the provision of ubiquitous services in the context of the Internet of Things.

Initially we have evaluated the Internet protocols that provide presence information and we have performed an analysis and evaluation regarding its implementation in the WSN considering the limitations and specific characteristics of these networks such as bandwidth, processing capacity and energy. The results indicated that existing Internet protocols for presence service could not be implemented directly in the WSN because they are not focused on energy efficiency and optimization communication which are important aspect that impact WSN resources. Therefore, we have identified the need for a protocol that considers these important aspects and we proposed several key points such as reliability, sleeping support and messaging format to consider for a presence service protocol on WSN.

Considering that presence service technologies are based on Publish/Subscribe model, we continued with the analysis and evaluation of existing Publish/Subscribe protocol on WSN. We proposed a presence service architecture for WSN based on Publish/Subscribe model because the most of protocols in the reviewed literature lack of mechanism such as data aggregation, reliability and energy efficiency to provide this service. The results of evaluation of our proposal showed that the use of a distributed network architecture composed by broker domains results in improving scalability, flexibility and reducing the network bottleneck. In addition, the publication on demand mechanism and the use of our proposed data aggregation algorithm provided an efficient energy management. The first one only allows to transmit publication message if there are related subscriber nodes. The last one reduces the traffic load in conditions of network congestion. We applied all proposed mechanisms of our architecture by designing a system approach called PASH addressed to home control based on Ambient Assisting Living concept. With PASH system we demonstrated the scalability and flexibility provided by Publish/Subscribe model. We also achieved distribute the intelligence among smart objects or devices and we also isolated the traffic by room or home area using broker domains. We concluded in this part, that presence information provided by WSN nodes embedded on devices or objects can be used to perform action to facilitate the home living.

In the rest of this thesis we focused on proposing several mechanisms to provide QoS features such as reliability and timeliness in delivery of packets for publish/subscribe protocols on WSN. These mechanisms could become of paramount importance in the design of applications that require to receive a message to react on time or in real-time to an event.

The most of protocols in the reviewed literature use reliability mechanisms in the delivery of packets based on a fixed packet retransmission mechanism such as MQTT-SN and CoAP using its default congestion control mechanism. We argued that this is an important drawback because they do not consider the conditions of the network in which they operate to calculate an RTO (Retransmission Timeout).

Based on this situation, we initially evaluated both protocols and proposed a new and simple adaptive retransmission mechanism using a parameter K and considering the network RTT that represent the conditions of the network to respond to the loss of packets in the most

appropriate manner. We also evaluated, a later CoAP proposal that considered an adaptive RTO calculation based on RFC 6298 which was also compared with our proposal. The results showed that our adaptive RTO mechanism provides an increase in PDR for each subscriber node compared with MQTT-SN and CoAP. Moreover, the results showed the effect of the chosen K value is mainly on the packet delivery ratio and the discarded publication ratio. We found the setting for obtaining the highest PDR for subscriber nodes, mainly for the node receiving publication messages in reliable mode. We conclude that in general, an increase in the K value yields a higher PDR and a lower discarded publication ratio and we demonstrated that we can obtain an optimized K value for each evaluated scenario to adapt to network conditions. However, it is also necessary to consider that this could lead to spurious retransmissions, and thus to duplicated messages. Finally, evaluation of the results showed that the RTO calculation plays an important role in all the metrics evaluated and there is a trade-off between PDR and the retransmitted message ratio, depending on the K value. Therefore, the investigation demonstrates that we can achieve better performance by using the optimized K value.

To cope the different requirements of application and to provide a way to express and negotiate QoS features to the applications, we proposed finally a mechanism that establishes three different QoS levels to providing mainly reliability (QoS level 1), reliability with data aggregation (QoS level 2) and timeliness (QoS level 3) on packet delivery based on Publish/Subscribe model for wireless sensor networks.

For the QoS level 1 providing reliability on packet delivery, we enhanced our previous adaptive RTO mechanism by adjusting the K parameter value depending on calculated PDR on each verification window for the subscriber node. We evaluated and compared our new adaptive RTO mechanism and the CoCoA mechanism of CoAP protocol and the results showed a better performance in the obtained PDR by our proposal.

In the case of QoS level 2, we demonstrated that providing reliability with data aggregation is useful in network congestion conditions because at this moment, channel occupation is reduced thus the other subscriber nodes take advantage to receive publication packets and to access the channel with less contention to send its confirmation packets. With the presented results we concluded that the simple aggregation mechanism could help to reduce congested situations and to improve performance in the packet delivery. However, we consider this QoS level is oriented to messages that are not critical in the “arrival on time” target to the destination node, so that there is no conflict with real time sensor readings.

For timeliness on packet delivery, we proposed initially a mechanism to provide priority support for Observe model of CoAP protocol based on four QoS levels for delivering updates. The results showed that this mechanism decreases the experimented delay and saves bandwidth and energy because each observer would only receive the updates based on its role in the application and the subject would avoid sending updates to all nodes in the network. After this contribution, we proposed a new mechanism to provide timeliness based on deadline mechanism which select the packets that have a higher probability to achieve the subscriber node based on the defined deadline target. This mechanism considered the experienced RTT as a metric to obtain the network condition and take a decision to send a message to meet the defined deadline. The results showed that packets sent to subscriber nodes were received before the defined deadline using our proposal also we achieve a reduction of energy consumption because broker node would discard every packet not accomplish with the deadline target. In this QoS level we are not concerned on subscriber PDR obtained, because it could be lower as more packets are discarded.

In this thesis we have contributed in the development of mechanisms that allow the Publish / Subscribe protocols to comply with different requirements of the WSN applications. However, there are still some aspects that must be studied and analyzed in greater depth and that we have defined as future work.

- The enhancing of our adaptive RTO mechanism. It could consider the receiving messages rate at the broker node, the number of publisher nodes registered in the network to calculate the network load and also the number of hops to the destination. We thereby expect to gain control of duplicated messages that lead mainly to energy waste, thus increasing the PDR of subscriber nodes. This is the reason of the trade-off observed in the results of this study between PDR and energy consumption.

- The analysis and evaluation of the effect of the routing protocol in the performance of our proposal. Besides, it could propose a cross layer architecture to get information from the others layers to enhance the adjusting of the RTO to obtain a better PDR and timeliness on packet delivery and also get energy and bandwidth efficiency.

- The analysis and evaluation of the effect of adapting TCP protocol mechanism in Publish/Subscribe protocols to integrate WSN with Internet infrastructure.

References

Contributions

Journal papers:

[P1]. García Davis, E., Calveras, A., Demirkol, I. Improving Packet Delivery Performance of Publish/Subscribe Protocols in Wireless Sensor Networks. Sensors 2013, 13(1), pp. 648-680. (IF 2013 = 2.048, Q1).

[P2]. Davis, E. G. and Augé, A. C. 2018. Publish/Subscribe Protocol in Wireless Sensor Networks: Improved Reliability and Timeliness. KSII Transactions on Internet and Information Systems 2018, 12(4), pp.1527-1552. DOI: 10.3837/tiis.2018.04.008. (IF 2017 = 0.611, Q4).

Conference papers:

[P3]. García, E., Calveras, A. Presence Service for Wireless Sensor Networks: Research and Open Issues. In proceedings of the 9th conference of telematics engineering, Valladolid, Spain, 29 September – 1 October 2010.

[P4]. García, E., Calveras, A. Presence-Based Architecture for Wireless Sensor Networks Using Publish/Subscribe Paradigm, the 9th IFIP TC 6 International Conference on Wired/Wireless Internet Communications - WWIC 2011, LNCS Vol. 6649, pp. 27-38. Barcelona (Spain) June 2011. DOI: 10.1007/978-3-642-21560-5_3

[P5]. García, E., Calveras, A. A Presence-aware Smart Home System (PASH), III International Workshop on Ambient Assisted Living - IWAAL 2011 held at IWANN 2011, LNCS Vol. 6693, pp. 159-166. Barcelona (Spain) June 2011. DOI: 10.1007/978-3-642-21303-8_22

[P6]. Ludovici, A.; Garcia, E.; Gimeno, X.; Calveras Auge, A., Adding QoS support for timeliness to the observe extension of CoAP," IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), pp.195-202, Barcelona, Spain, 8-10 Oct. 2012

Bibliography

[1]. Alemdar, H., Ersoy, C. Wireless sensor networks for healthcare: A survey Computer Networks, Volume 54(15), October 2010, Page(s) 2688-2710.

[2]. Escolar Díaz, S. et al. A novel methodology for the monitoring of the agricultural production process based on wireless sensor networks. Computers and Electronics in Agriculture, Volume 76(2), May 2011, Page(s) 252-265

[3]. Tacconi, D., et al. Using wireless sensor networks to support intelligent transportation systems. Ad Hoc Networks, Volume 8(5), July 2010, Page(s): 462-473

[4]. Han, D., Lim, J. Smart home energy management system using IEEE 802.15.4 and zigbee. IEEE Transactions on Consumer Electronics. Volume: 56(3) 2010, Page(s): 1403 – 1410.

[5]. Lennvall, Tomas; Gidlund, Mikael; Åkerberg, Johan. Challenges when bringing IoT into Industrial Automation. 2017 IEEE AFRICON. pp 905-910 DOI: 10.1109/AFRCON.2017.8095602

[6]. Burange A.W; Misalkar H.D.; Review of Internet of Things in Development of Smart Cities with Data Management & Privacy. IEEE Conf. International Conference on Advances in Computer Engineering and Application (ICACEA), 2015, 189 - 195, DOI: 10.1109/ICACEA.2015.7164693

[7]. Kehua Su; Jie Li; Hongbo Fu. Smart city and the applications. International Conference on Electronics Communications and Control (ICECC), 2011, pp. 1028-1031.

[8]. Yan, Hairong; Huo, Hongwei; Xu, Youzhi; Gidlund, Mikael. Wireless sensor network based e-health system - implementation and experimental results. IEEE Transactions on Consumer Electronics. Volume: 56, Issue: 4, November 2010. Pp. 2288-2295.

[9]. Day, M., Rosenberg, J., Sugano, H.; A Model for Presence and Instant Messaging; RFC 2778; The Internet Society; February 2000.

[10]. Dietrich, I., Dressler, F. On the lifetime of wireless sensor networks. ACM Transaction Sensor Networks Volume 5(1) February 2009. ACM New York, NY, USA ISSN: 1550-4859.

[11]. Eugster, PT.: The many faces of publish/subscribe. ACM Computing Surveys. June 2003.

[12]. B., Russello, G., Mostarda, L., Dulay, N. A policy-based publish/subscribe middleware for sense-and-react applications. Journal of Systems and Software, Volume 84(4), April 2011, Page(s) 638-654

[13]. [C] Wu, S., Zhang, W., Wang, H. PS4WSN- a publish/subscribe middleware for wireless sensor networks. IEEE International Conference on Information and Automation (ICIA), 2010. Page(s): 2242-2247, 20-23 June 2010.

[14]. Tran, Duc A. et al. Publish/Subscribe Techniques for Sensor Networks. May 2009.

[15]. Choon-Sung Nam; Hee-Jin Jeong; Dong-Ryeol Shin; Design of wireless sensor networks middleware using the publish/subscribe paradigm; IEEE International Conference on Service Operations and Logistics, and Informatics, IEEE/SOLI 2008(1), 2008, Page(s): 559 – 563.

[16]. Atzori, L., Iera, A., Morabito, G.; The Internet of Things: A survey. Computer Networks, 54(15): Page(s): 2787–2805, October 2010.

[17]. Rosenber, J.; A Presence Event Package for the Session Initiation Protocol (SIP); RFC 3856; The Internet Society; August 2004.

[18]. Saint-Andre, P.; Extensible Messaging and Presence Protocol (XMPP): core; Request for Comments 6120 (RFC 6120) The Internet Society, March 2011.

[19]. Bhuyan, B; Deva Sarma, H; Sarma, N.; Kar, A.; Mall, R. Quality of Service (QoS) Provisions in Wireless Sensor Networks and Related Challenges. Wireless Sensor Network 2(11), November 2010. pp. 861-868.

[20]. Z. Shelby, K. Hartke, C. Bormann, The Constrained Application Protocol (CoAP), Request for Comments 7252 (RFC7252), 2014.

[21]. K. Hartke. Observing Resources in the Constrained Application Protocol (CoAP). *Request for Comments (RFC7641)* 2015

[22]. Urs Hunkeler, Hong Linh Truon." MQTT-S — A publish/subscribe protocol for Wireless Sensor Networks," *3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE).* pp 791-798, 2008.

[23]. C. Bormann. A. Betzler, C. Gomez, I. Demirkol, "CoAP Simple Congestion Control/Advanced. draft-bormann-core-cocoa" 2018. Available on: https://tools.ietf.org/html/draft-bormann-core-cocoa (Accesed January 2019).

[24]. P.N. Renjith, E. Baburaj, "An analysis on data aggregation in Wireless Sensor Networks," International Conference on Radar, Communication and Computing (ICRCC), pp 62- 71, 2012

[25]. Sugano, H. et al. Presence Information Data Format. PIDF; *Request for Comments 3863 (*RFC 3863); The Internet Society; August. 2004.

[26]. Schulzrinne, H. et al. Rich Presence Extensions to the PIDF; *Request for Comments* 4480 (RFC 4480); The Internet Society; July 2006.

[27]. Schulzrinne, H. et al. Contact Information for the PIDF; *Request for Comments 4482 (*RFC 4482); The Internet Society; July 2006.

[28]. Peterson, J. GEOPRIV Location Object Format. *Request for Comments 4119 (*RFC 4119); The Internet Society; December 2005.

[29]. Barachi, M.; Kadiwal, A.; Glitho, R.; Khendek, F.; Dssouli, R.; A Presence-Based Architecture for the Integration of the Sensing Capabilities of Wireless Sensor Networks in the IP Multimedia Subsystem; IEEE Wireless Communications and Networking Conference. WCNC 2008; Page(s): 3116 – 3121

[30]. Barachi, M.; Kadiwal, A.; Glitho, R.; Khendek, F.; Dssouli, R.; The Design and Implementation of Architectural Components for the Integration of the IP Multimedia Subsystem and Wireless Sensor Networks; IEEE Communications Magazine; Volume 48(4) 2010; Page(s): 42-50.

[31]. IEEE Std 802.15.4-2003‖, IEEE 802.15 WPANTM Task Group 4, 2003, http://www.ieee802.org/15/pub/TG4.html

[32]. Millard, P., Saint-Andre, P., Meijer, R.; XEP-0060: Publish-Subscribe; XMPP Standards Foundation; 2010.

[33]. Saint-Andre, P., Smith, K.; XEP-0163: Personal Eventing Protocol; XMPP Standards Foundation; 2010.

[34]. Gómez, C., Paradells J., Caballero, J. E.; Sensor Everywhere: Wireless Network Technologies and Solutions; Fundación Vodafone España, ISBN 978-84-934740-5-8

[35]. Augeri, C.J., Bulutogiu, D.A., Mullins, B.E., Baldwin, R.O., Blaird, LC; An analysis of XML Compression efficiency; In Proceedings of the 2007 Workshop on Experimental Computer Science, ExpCS 2007, ACM June 2007.

[36]. Kamiya, T., Schneider, J.; Efficient XML Interchange (EXI) Format 1.0; World Wide Web Consortium Last Call WD-exi-20080919, September 2008.

[37]. Open Geospatial Consortium, Binary Extensible Markup Language (BXML) Encoding Specification, Version 0.08; January 2006.

[38]. Kuorilehto, M., et alt. Experimenting TCP/IP for Low-Power Wireless Sensor Networks. The 17th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'06). Sept. 11-14 2006. Page(s) 1-6

[39]. Wang, H., Xiong, D., Wang, P.,Liu, Y. A Lightweight XMPP Publish/Subscribe Scheme for Resource-Constrained IoT Devices. IEEE Access, 5 2017, pp.16393-16405.

[40]. Klauck, R. & Kirsche, M. Chatty things - Making the Internet of Things readily usable for the masses with XMPP. In C. Pu, J. Joshi & S. Nepal (eds.), CollaborateCom (p./pp. 60-69), ICST / IEEE 2012. ISBN: 978-1-4673-2740-4

[41]. A. Hornsby and E. Bail, "uXMPP: Lightweight Implementation for Low Power Operating System Contiki," in Proceedings of the International Conference on Ultra-Modern Telecommunications and Workshops (ICUMT 2009). IEEE, Oct. 2009, pp. 1-5.

[42]. Desai, P., Sheth, A., Anantharam, P. Semantic Gateway as a Service architecture for IoT Interoperability. IEEE International Conference on Mobile Services IEEE 2015. pp. 313-319.

[43]. Pawara, S.R., Hiray, S.R. Instant Notification System in Heterogeneous Sensor Network with Deployment of XMPP Protocol. International Conference on Cloud & Ubiquitous Computing & Emerging Technologies IEEE 2013. pp. 87-92

[44]. Krishnamurthy, S. TinySIP: Providing Seamless Access to Sensor-based Services. 3rd Annual International Conference on Mobile and Ubiquitous Systems - Workshops, July 2006.

[45]. Luckenbach, T., Gober, P., Arbanowski, S., Kotsopoulos, A; TinyREST: A protocol for integrating sensor networks into Internet; In Proc. Of REALWSN 2005.

[46]. Fielding, R.T., Taylor, R.N.; Principled design of the modern Web architecture; Proceedings of the 2000 International Conference on Software Engineering ICSE 2000, Limerick, Ireland, June 2000, Page(s): 407-416.

[47]. Adi, V., Padmanabh, K., Sanjoy, P.; senSebuddy: A Buddy to Your Wireless Sensor Network; Real-World Wireless Sensor Networks Lecture Notes in Computer Science, 2010, Volume 6511/2010, Page(s): 106-112.

[48]. Barachi, M.; Kadiwal, A.; Glitho, R.; Khendek, F.; Dssouli, R.; A Presence-Based Architecture for the Integration of the Sensing Capabilities of Wireless Sensor Networks in the IP Multimedia Subsystem; IEEE Wireless Communications and Networking Conference. WCNC 2008; Page(s): 3116 – 3121

[49]. Barachi, M.; Kadiwal, A.; Glitho, R.; Khendek, F.; Dssouli, R.; The Design and Implementation of Architectural Components for the Integration of the IP Multimedia Subsystem and Wireless Sensor Networks; IEEE Communications Magazine; Volume 48(4) 2010; Page(s): 42-50.

[50]. Gomez C.; Arcia-Moret, A; Crowcroft, J.; TCP in the Internet of Things: From ostracism to prominence. IEEE Internet Computing; Volume 22(1) 2018; Page(s): 29-41.

[51]. Esposito, C., Cotroneo, D., Russo, S. On reliability in publish/subscribe services. Computer Networks. Volume 57, Issue 5, 7 April 2013, Pages 1318-1343.

[52]. Russello, G., Mostarda, L., Dulay, N. A policy-based publish/subscribe middleware for sense-and-react applications. Journal of Systems and Software, Volume 84(4), April 2011, Page(s) 638-654

[53]. Wu, S., Zhang, W., Wang, H. PS4WSN- a publish/subscribe middleware for wireless sensor networks. IEEE International Conference on Information and Automation (ICIA), 2010. Page(s): 2242-2247, 20-23 June 2010.

[54]. Tran, Duc A. et al. Publish/Subscribe Techniques for Sensor Networks. May 2009.

[55]. Souto, E., et alt. Mires: a publish/subscribe middleware for sensor networks. Personal and Ubiquitous Computing. Vol. 10, Feb 2006. Pp 37-44

[56]. Cugola, G., H.-Arno Jacobsen; Using Publish/Subscribe Middleware for Mobile Systems; ACM SIGMOBILE Mobile Computing and Communications Review; Volume 6 Issue 4, October 2002, Page(s) 25-33, ACM New York, NY, USA ISSN: 1559-1662

[57]. Fidler, E., Jacobsen, H-A., Li, G., Mankovski, S.; The Padres distributed publish/subscribe system. In Proc. of 8th International Conference on Feature Interactions in Telecommunications and Software System, 2005.

[58]. Hall, C.P., Wolf, A., Carzaniga, A., Rose, J., Wolf, E.L.; A content-based networking protocol for sensor networks; Department of Computer Science, University of Colorado, Tech Rep. August 2004.

[59]. Costa, P., Picco, G. Semi-probabilistic Content-based Publish/Subscribe. In Proc. of the 25th Int. Conf. on Distributed Computing Systems (ICDCS05), page(s) 575–585, Columbus, USA, June 2005. IEEE Computer Society Press.

[60]. Intanagonwiwat, C., Govindan, R., Estrin, D.; Directed diffusion: a scalable and robust communication paradigm for sensor networks. In MobiCom ’00: Proceedings of the 6th annual international conference on Mobile computing and networking, pages 56–67. ACM Press, 2000.

[61]. Fiege, L., Cilia, M., Muhl, G., Buchmann, A.; Publish-Subscribe grows up: Support for management, visibility control, and heterogeneity. IEEE Internet Computing, Volume 10(1), Page(s): 48-55. 2006.

[62]. Y. Huang and H. Garcia-Molina. Publish/subscribe tree construction in wireless ad-hoc networks. In MDM ’03: Proceedings of the 4th International Conference on Mobile Data Management, pages 122–140, London, UK, 2003. Springer-Verlag.

[63]. J. H. Schonherr, H. Parzyjegla, and G. M¨uhl. Clustered publish/subscribe in wireless actuator and sensor networks. In MPAC ’08: Proceedings of the 6th international workshop on Middleware for pervasive and ad-hoc computing, page(s) 60–65, New York, NY, USA, 2008. ACM.

[64]. Avin, C., Krishnamachari, B.; The power of choice in random walks: An empirical study. Computer Networks, 52(1) page(s):44–60, 2008.

[65]. Braginsky, D., Estrin, D.; Rumor routing algorithm for sensor networks. In WSNA ’02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, page(s) 22–31. ACM, 2002.

[66]. Jaime Chen, Manuel Diaz, Bartolome Rubio, Jose Troya, “PS-QUASAR: A publish/subscribe QoS aware middleware for Wireless Sensor and Actor Networks,” Journal of System and Software, vol. 86, no. 6, pp. 1650-1662, 2013.

[67]. Hui-Ling Chang, Cheng-Gang Wang, Mong Ting Wu, Meng-Hsun Tsaiet, Chia-Ying Lin, “Gateway-Assisted Retransmission for Lightweight and Reliable IoT Communications,” Sensors, vol. 16 no. 10, 2016

[68]. Pruet Boonma, Junichi Suzuki, “TinyDDS: an interoperable and configurable publish/subscribe middleware for wireless sensor networks,” Principles and Applications of Distributed Event-Based Systems, pp. 26, 2010.

[69]. Kai Beckman, Olga Dedi, “sDDS: a portable data distribution service implementation for WSN and IoT platforms,”. In Proc. of 12th Int. Workshop on Intelligent Solutions in Embedded Systems (IEEE WISES), pp. 115-120, 2015.

[70]. Xiaoyu Tong, Edith Ngai, “A Ubiquitous Publish/Subscribe Platform for Wireless Sensor Networks with Mobile Mules,” IEEE 8th International Conference on Distributed Computing in Sensor Systems (DCOSS). pp. 99-108, 2012.

[71]. Ramesh, R., Pramod, K. V.; Data aggregation techniques in sensor networks: A survey; IEEE Comm. Surveys & Tutorials; 2006.

[72]. Ajay. K. Talele; Suraj G. Patil; Nilkanth. B. Chopade A Survey on Data Routing and Aggregation Techniques for Wireless Sensor Networks. IEEE International Conference on International Conference on Pervasive Computing (ICPC). Pp. 1-5. 2015.

[73]. Ghaffariyan, P. An Effective Data Aggregation Mechanism for Wireless Sensor Networks. 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), 2010, Page(s): 1 – 4.

[74]. Poorter, E. et al. Non-intrusive aggregation in wireless sensor networks. Ad Hoc Networks, Volume 9(3), May 2011, Page(s) 324-340.

[75]. M Schiefer, C Steup, J Kaiser. “Real world testing of aggregation in publish/subscribe systems”. IEEE International Conference on Wireless Information Networks. 2013.

[76]. Mohsen Sharifi, Majid Alkaee Taleghan, Amirhosein Taherkordi, “A Publish-Subscribe Middleware for Real-Time Wireless Sensor Networks,” International

Conference on Computational Science; ICCS 2006, LNCS, 2006, vol. 3991/2006, pp. 981-984, 2006.

[77]. Mohsen Sharifi, Majid Alkaee Taleghan, Amirhosein Taherkordi, "A Middleware Layer Mechanism for QoS Support in Wireless Sensor Networks", in Proc. of the Fifth International Conference on Networking (ICN), 2006.

[78]. Felemban E., Lee C. G., Ekici E., Boder R. and Vural S., Probabilistic QoS guarantee in reliability and timeliness domains in wireless sensor networks, Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies., 2005, pp. 2646-2657 vol. 4. doi: 10.1109/INFCOM.2005.1498548

[79]. Felemban E., Lee C. G and Ekici E., MMSPEED: multipath Multi-SPEED protocol for QoS guarantee of reliability and. Timeliness in wireless sensor networks, in IEEE Transactions on Mobile Computing, vol. 5, no. 6, pp. 738-754, 2006. doi: 10.1109/TMC.2006.79

[80]. Dobslaw F., Zhang T. and Gidlund M., QoS-Aware Cross-Layer Configuration for Industrial Wireless Sensor Networks, in IEEE Transactions on Industrial Informatics, vol. 12, no. 5, pp. 1679-1691, 2016. doi: 10.1109/TII.2016.2576964

[81]. Hasan M. Z.; Al-Rizzo H.; Al-Turjman F., A Survey on Multipath Routing Protocols for QoS Assurances in Real-Time Wireless Multimedia Sensor Networks, in IEEE Communications Surveys & Tutorials, vol. 99, pp.1-1 2017 doi: 10.1109/COMST.2017.2661201

[82]. K. Hartke. Observing Resources in the Constrained Application Protocol (CoAP). Request for Comments (RFC7641) 2015

[83]. OASIS Foundation. MQTT standard version 3.1.1 October 29, 2014. Available online: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.pdf. (accessed January 2019).

[84]. OASIS Foundation. MQTT-SN protocol specification version 1.2. November 14, 2013. Available online: http://mqtt.org/new/wp-content/uploads/2009/06/MQTT-SN_spec_v1.2.pdf (accessed January 2019).

[85]. Cam, H., Sahingoz, O.K., Sonmez, A.C. Wireless sensor networks based on publish/subscribe messaging paradigms. In: Proceedings of the 6th International Conference on Advances in Grid and Pervasive Computing, (GPC) 2011. Springer-Verlag, Berlin, Heidelberg, pp. 233–242 90

[86]. Andréu J., Viúdez J., Holgado J.A. (2009) An Ambient Assisted-Living Architecture Based on Wireless Sensor Networks. In: Corchado J.M., Tapia D.I., Bravo J. (eds) 3rd Symposium of Ubiquitous Computing and Ambient Intelligence 2008. Advances in Soft Computing, vol 51. Springer, Berlin, Heidelberg. pp. 239-249.

[87]. G. Zhao. Wireless Sensor Networks for Industrial Process Monitoring and Control: A Survey. Network Protocols and Algorithms. Vol 3(1), 2011. pp. 46-63.

[88]. OMNet++ Simulator. Available online: http://www.omnetpp.org (accessed on January 2019). 87

[89]. Paxson V. Allman M. Chu J. Sargent M. Computing TCP's Retransmission Timer. Request for Comments: 6298 (RFC6298) 2011 88

[90]. Telosb Datasheet. Available online: http://www.memsic.com/userfiles/files/Datasheets/WSN/telosb_datasheet.pdf (accessed on January 2019) 89

[91]. Burange A.W; Misalkar H.D.; Review of Internet of Things in Development of Smart Cities with Data Management & Privacy. IEEE Conf. International Conference on Advances in Computer Engineering and Application (ICACEA), 2015, 189 - 195, DOI: 10.1109/ICACEA.2015.7164693

[92]. Kehua Su; Jie Li; Hongbo Fu. Smart city and the applications. International Conference on Electronics Communications and Control (ICECC), 2011, pp. 1028-1031.

[93]. Gama Ó., Carvalho P., Afonso J.A., Mendes P.M. (2009) Quality of Service in Wireless e-Emergency: Main Issues and a Case-Study. In: Corchado J.M., Tapia D.I., Bravo J. (eds) 3rd Symposium of Ubiquitous Computing and Ambient Intelligence 2008. Advances in Soft Computing, vol 51. Springer, Berlin, Heidelberg

[94]. Zolertia Z1 Datasheet. Available Online: http://zolertia.sourceforge.net/wiki/images/e/e8/Z1_RevC_Datasheet.pdf (accessed on January 2019) 91

[95]. A. Nasipuri, R. Cox, J. Conrad, L. Van der Zel, B. Rodriguez and R. McKosky, "Design considerations for a large-scale wireless sensor network for substation monitoring," *IEEE Local Computer Network Conference*, Denver, CO, 2010, pp. 866-873.