

A Fair and Distributed Congestion Control Mechanism for Smart Grid Neighborhood Area Networks

Juan Pablo Astudillo León, Thomas Begin, Anthony Busson, Luis Javier de La Cruz Llopis

▶ **To cite this version:**

Juan Pablo Astudillo León, Thomas Begin, Anthony Busson, Luis Javier de La Cruz Llopis. A Fair and Distributed Congestion Control Mechanism for Smart Grid Neighborhood Area Networks. Ad Hoc Networks, 2020, 104, pp.1-13. 10.1016/j.adhoc.2020.102169 . hal-02563269

HAL Id: hal-02563269
https://hal.science/hal-02563269

Submitted on 14 May 2020

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

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

Available online at www.sciencedirect.com
ScienceDirect
Ad Hoc Networks 00 (2020) 1–21

A Fair and Distributed Congestion Control Mechanism for Smart Grid Neighborhood Area Networks

Juan Pablo Astudillo León[a,*], Thomas Begin[b], Anthony Busson[b], Luis J. de la Cruz Llopis[a]

[a]*Department of Network Engineering, Universitat Politècnica de Catalunya (UPC), Barcelona 08034, Spain*
[b]*Univ Lyon, Université Claude Bernard Lyon 1, ENS de Lyon, Inria, CNRS, LIP, France*

Abstract

The need for significant improvements in the management and efficient use of electrical energy has led to the evolution from the traditional electrical infrastructures towards modern Smart Grid networks. Taking into account the critical importance of this type of networks, multiple research groups focus their work on issues related to the generation, transport and consumption of electrical energy. One of the key research points is the data communication network associated with the electricity transport infrastructure, and specifically the network that interconnects the devices in consumers' homes, the so-called Neighborhood Area Networks (NANs). In this paper, a new fair and distributed congestion control mechanism for NANs is proposed, implemented and evaluated. The main goal of this mechanism is to provide fairness in the access to the network, thus avoiding that some network nodes monopolize the use of the channels due to their higher traffic generation rate, or to their geographical position. Besides, different priorities have been considered for the traffic flows transmitted by different applications. The goal here is to provide the needed Quality of Service (QoS) to all traffic flows, especially when the traffic load is high. The proposal is evaluated in the context of a wireless ad hoc network composed by a set of smart grid meter devices. Applying our proposed congestion control mechanism leads to performance improvements in terms of packet delivery ratio, network throughput fairness between different traffic sources, packet network transit time and QoS provision.

Keywords:
Smart Grid, Neighborhood Area Networks, Congestion Control, Fairness, Traffic Differentiation, QoS

1. Introduction

Electricity is an essential resource widely used in our daily activities. Taking into account the importance of this resource, research groups have focused their efforts to improve the management and operation of the traditional electricity distribution networks. To this end, in Smart Grids a data communication network has been incorporated into the electric infrastructure. The objective is to provide new services to both the supplying companies and to their customers [1]. In a society where new technologies are growing fast, a large number of new services can be offered by the smart grid data communication network. Generally speaking, most smart grid applications have strong security and reliability requirements. Therefore, keeping the desired Quality of Service (QoS) for both classical and emerged applications is of paramount importance when planning and operating this type of networks [2].

*Corresponding author. Tel.: +34-93-401-60-14.
Email addresses: juan.pablo.astudillo@upc.edu (Juan Pablo Astudillo León), thomas.begin@univ-lyon1.fr (Thomas Begin), anthony.busson@univ-lyon1.fr (Anthony Busson), luis.delacruz@upc.edu (Luis J. de la Cruz Llopis)

The smart grid data communication network is divided into three different sub-networks: Home Area Network (HAN), Neighborhood Area Network (NAN) and Wide Area Network (WAN) [3]. On the one hand, the HAN interconnects the Smart Meters (SM) and other possible appliances present inside the homes. Possible standards for HANs include IEEE 802.15.4 (Low-Rate Wireless Personal Area Networks, LR-WPAN) and the traditional IEEE 802.11 (Wireless Local Area Networks, WLAN). In turn, HANs are interconnected through the NANs. Here, selectable technologies are the well-known Power Line Communication (PLC), or wireless standards such as IEEE 802.15.4g, IEEE 802.11s and also WLAN. Finally, the WAN allows the information exchange between the different NANs and the control center. A wired backbone or a wireless technology such as Wireless Metropolitan Area Networks (Wireless MAN, IEEE 802.16) can be considered.

The goal of this paper is to present and evaluate a distributed congestion control mechanism that improves the performance offered by the NANs, when the selected technologies are IEEE 802.11 Wireless Ad Hoc Network (WANET) and IEEE 802.11ac physical layer standard. Our solution is conceptually simple as well as easily tunable and implementable. In addition, it is agnostic to the routing protocol, and to the MAC and physical layers. The proposed mechanism combines several algorithms that can differentiate the quality of service offered to each traffic based on their criticality, while providing a fair service to all nodes in the network. Note that our proposed solution can be viewed as a middleware candidate for smart grids that performs over the network infrastructure with the aim of ensuring a fair sharing of the resources between nodes and traffic classes as well as high levels for resource utilization and packet delivery ratio.

The rest of the paper is organized as follows. Section 2 discusses related work. The system under study and the proposed solution are presented in Sections 3 and 4, respectively. Its performance is evaluated in Section 5. Finally, Section 6 concludes the paper.

2. Related work

Several researchers have focused their efforts with the aim of improving the performance offered by the smart grid neighborhood area networks. For instance, authors in [4] present a performance evaluation and comparison of Optimized Link State Routing Protocol (OLSR) [5] and HWMP (Hybrid Wireless Mesh Protocol defined in IEEE 802.11s) [6] routing protocols, where a classification of the main AMI (Advanced Metering Infrastructure) application traffics is taking into account. Later, in [7], the same authors propose an enhancement of the OLSR protocol by using the combination of different basic metrics. They use Relevant Link Metric Types (RLMTs) for each application, together with the AHP (Analytical Hierarchy Process) algorithm in order to select the best path. Based on IEEE 802.11s, a multigate communication network is proposed in [8] for improving the network performance for smart grids. They present a multi-gateway, multi-channel and real-time traffic scheduling. On the other hand, to ensure the needed QoS of different smart grid applications, the authors of [9] propose a modification to the HWMP protocol (HWMP-NQ). They also present a modification of the airtime link metric by considering the packet size and the transmission rate. Finally, they present the benefits of a multi-gateway backup routing, and a modification of the path error mechanism to reduce the routing overhead.

Given that most data traffic is transmitted upstream from the smart meters to the data concentrator, authors in [10] present modifications to the airtime link metric calculation and to the path selection mechanism, in order to provide greater relevance to the upstream transmissions. Besides, authors address the need for congestion control when the network size is large. Later, some of the authors of [10] analyze in [11] the reliability and the weaknesses of the HWMP protocol when it is integrated to smart grid networks. Authors propose some solutions based again on modifications to the airtime link metric and in the possibility of routes reservation. Besides, they present a novel delay-tolerant traffic management method in order to provide a better quality of service to some applications.

A different modification of the HWMP protocol was presented in [12] that allows an efficient selection of paths among multiple possibilities, depending on the quality of service requirements of the different traffic flows. Also, the mechanism is complemented with the assignment of different frequency channels to each available path. The main goal is to transmit priority traffics through the best paths, and using also the less congested channel. Besides, different channels are used for data and control traffic. Later, in [13], a multi-channel allocation scheme together with a congestion control mechanism was presented and evaluated, considering the different service quality needs of the different applications. The congestion control mechanism takes into account if the network is in an emergency state caused by intrinsic or extrinsic reasons. Different congestion control functions are proposed for each traffic, which

provide a higher transmission probability to the traffic with higher QoS needs, based on the network congestion level and on the emergency state.

In the current paper, a different solution to improve the network performance for NANs is proposed, implemented and evaluated. Instead of considering IEEE 802.11s that is designed for mesh networks, we consider the ad hoc mode of 802.11. Note that although our solution does not rely on a specific IEEE 802.11 physical layer, we consider those included in the recent 802.11ac standard for our numerical evaluation.

On the other hand, unlike the strategy followed in [12] and [13] where relay nodes are in charge of discarding packets in network congestion situations, in this paper it is up to the source nodes to reduce their packet generation rate. Thus, depending on the sensed degree of network congestion, source nodes may increase or decrease the rate at which they generate their packets. This way, we avoid unnecessary transmissions of packets that will be likely discarded later on their way to their destination. To this end, new signaling messages must be transmitted from relay nodes to source nodes. Besides, the proposed mechanism is also designed to provide a fair distribution of the available network resources between all the source nodes, avoiding a higher utilization by the nodes with higher packet generation rate or simply favorably located. Finally, our solution is independent of the routing protocol used.

Note that the current paper extends a preliminary solution presented in [14] in several ways. We include new scenarios, new performance metrics and a deeper analysis of the results. In particular, we study the fairness in the attained network throughputs taking into account the distance between the smart meters and the data concentrator. The goal is to demonstrate how our solution provides a fair sharing of the network resources among the nodes regardless of their position. Furthermore, our solution is compared with the Enhanced Distributed Channel Access (EDCA). EDCA is a link layer access control mechanism, which provides transmission priority to higher access categories. Finally, we design, configure and evaluate a new scenario including downstream data traffic from the control center all the way to network nodes.

3. System description

Infrastructure topology

In this work, we study the Neighborhood Area Network (NAN) part of a smart grid communication infrastructure. Typically, a NAN comprises a set of N nodes corresponding to Smart Meters (SM) plus a data concentrator node (aka sink) that acts as a gateway to the WAN segment. Smart meters collect information from their HAN segment (e.g., gas meter, water meter, electricity meter, other sensors) and exchange them with the data concentrator node directly or through other smart meters that serve as relay nodes to form a multi-hop communication network. This is illustrated by Figure 1.

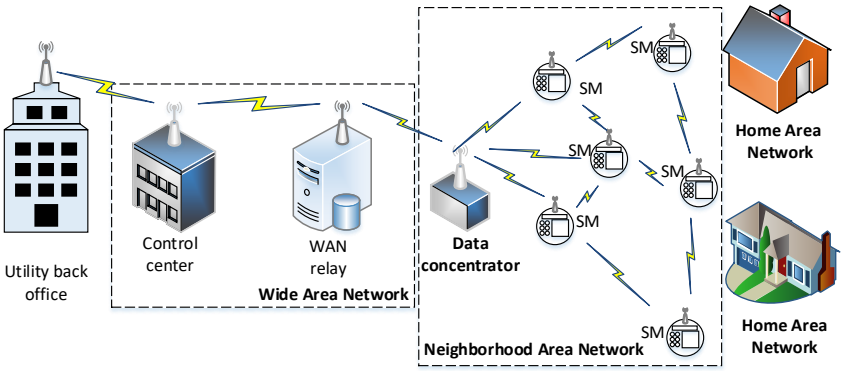


Figure 1: Representation of a Neighborhood Area Network with its smart meters (SM) and data concentrator.

Communication link

In our case, we consider that the nodes of the smart grid communicate using wireless transmissions. More precisely, we assume that the nodes implement the IEEE 802.11 standard. While this option comes with strong assets (e.g., unlicensed spectrum, off the shelf technology), its performance are typically hard to forecast as they strongly rely on the environment surrounding a node (viz., the quality of the radio channel as well as the number, location, and activity of its neighboring nodes).

Classes of data traffic

The data traffic sent over the NAN is diverse. In the most general case, it includes upstream flows (i.e., from the smart meters to the data concentrator) corresponding, for instance, to meter reading data, home energy and Electric Vehicle (EV) charging information, as well as downstream flows that may contain Demand Response information to home users to adjust, for instance, energy consumption during peak hours. Furthermore, traffic flows have typically different QoS requirements in terms of throughput and network transit time. More specifically, we assume that flows are divided into K different classes, numbered 1,...,K, where class 1 has the most stringent requirements. Each flow generates data at a given rate that we refer to as its nominal rate. This data rate is determined by the smart grid application and does not necessarily comply with the available communication resources of the NAN. We use x_n^k to denote the nominal rate (in bps) of the flow originating at node n ($n = 1, ..., N$) of class k ($k = 1, ..., K$) where N refers to the number of nodes. Finally, we indicate the number of source nodes by S. Clearly, we have: $S < N$.

Performance parameters

We study several metrics to evaluate the performance of the NAN as well as those attained by the flows present in the NAN. We consider the *channel utilization factor* on each node and we use ρ_n ($\in [0, 1]$) to denote its value at node n. More precisely, ρ_n refers to the occupation of the radio channel at node n averaged over a period of T_A seconds. We also evaluate the *Packet Delivery Ratio* (PDR) that returns, for each traffic (upstream) flow, the ratio of packets successfully received at the data concentrator node. We calculate the *attained throughput* that corresponds to the number of bytes per second that each traffic flow can successfully deliver to the data concentrator. Note that the values of PDR and attained throughput are generally studied jointly. We determine the *network transit time* for each traffic flow that amounts to the time needed by their packets to travel from their source to the data concentrator node. Besides, the compliance factor will also be studied. This factor shows the percentage of packets that have been delivered on time to their destination. Note that the maximum time allowed for packet delivery is typically different for each traffic class. Finally, we include in our analysis the *Jain's index* [15] calculated over the attained throughput for flows belonging to the same class but originating from different sources. The Jain's index is a common metric to assess the level of fairness between N entities. Its value ranges from $1/N$ (worst case) to 1 (best case). In our case, it is maximum when traffic flows attain the same throughput, regardless of their distance to the concentrator node.

Problem statement

From a high-level point of view, traffic flows with various QoS requirements and nominal rates occur over a NAN whose communication resources are not necessarily oversized. Therefore, there may be an utter need for auto-adapting the flow data rates so that they share the available communication resource in a consistent and controllable way.

Table 1: Principal notation.

Parameter	Description	Units
	System parameters	
N	Number of nodes	
K	Number of traffic classes	
S	Number of source nodes	
x_n^k	Nominal rate of the flow originating at node n of class k	bps
ρ_n	Current utilization of the radio channel at node n	
	Proposed solution parameters	
ρ_U	Over-utilization threshold for the radio channel	
ρ_L	Under-utilization threshold for the radio channel	
T_A	Periodicity of Algorithms 1 and 3	sec
y_j^k	Estimated rate for the flow originating at node j of class k	bps
β_I	Increasing factor in case of under-utilization	
β_D	Decreasing factor in case of over-utilization	
$z_{j,n}^k$	Adapted rate returned by node n for the flow originating at node j of class k	bps
δ^k	Unitary increment for class k	bps
α^k	Percentage (weight) assigned to class k	
s_j^k	Implemented flow rate for the flow originating at node j of class k	bps

4. Proposed FDCC solution

4.1. Distributed solution

Our proposed FDCC (Fair Distributed Congestion Control) solution comprises 3 algorithms and is distributed in the sense that it operates both at source and relay nodes. Bear in mind that a node (with the exception of the data concentrator) can be a relay, a source or both. The rationale of our algorithms is to enable high utilization of the radio channel while avoiding the occurrence of long-lasting congestions causing packet losses and collisions.

To achieve this trade-off, our solution assigns a certain portion of the (estimated) available communication resources to each flow based on their class and source. Note that the size of these portions are easily parameterized. Our solution breaks down as follows. Through Algorithm 1, relay nodes monitor their current utilization of the radio channel as well as the current rate for each flow hopping through them. Depending on the current value of radio channel utilization, Algorithm 2 is triggered, and delivers in return adapted source rates for each flow. Lastly, newly computed source rates are notified to the associated source nodes that readjust their flow rates as described in Algorithm 3.

Note that these latter notifications are the only control packets generated by our algorithms. Consequently, during each computation period (e.g., one second in our experiments), our solution generates at most a number of control packets equal to the number of relay nodes multiplied by the number of sources. Besides, executing the algorithms imply few instructions whose requirements in memory and CPU are low.

We now describe each of the three algorithms.

4.1.1. Detecting over- and under-utilization of the radio channel

Algorithm 1 is implemented at each relay node and monitors if the radio channel is over-utilized, or alternately, under-utilized. In practice, for each relay node n, the algorithm verifies if $\rho_L < \rho_n < \rho_U$ where ρ_L and ρ_U are constant values in between 0 and 1. Additionally, through Algorithm 1, relay nodes evaluate the current rate for each flow hopping through them. We use y_j^k to denote the current rate calculated for the flow of class k and originating from node j. Note that y_j^k is computed as a simple arithmetic average over a period of time A and that, in general, y_j^k and x_j^k (nominal rate) differ.

If $\rho_n \leq \rho_L$ or $\rho_n \geq \rho_U$, Algorithm 1 triggers the readjustment of rate for the flows hopping through node n in an attempt to get the radio channel utilization ρ_n back in the interval $[\rho_L; \rho_U]$. More precisely, in case of over-utilization (resp. under-utilization), Algorithm 1 calls Algorithm 2 with the current values of y_j^k and a parameter β_D (resp. β_I) which is a constant between 0 and 1 (resp. larger than 1). Otherwise, if ρ_n, lies in between ρ_L and ρ_U, then Algorithm 1 remains idle.

Algorithm 1: Run at each relay node n to detect under- or over-utilization of the radio channel.

Input: $T_A, \rho_L, \rho_U, \beta_D, \beta_I$

while *True* **do**
  Update value of ρ_n and y_j^k
  if $\rho_n \geq \rho_U$ **then**
    Throttle flows: compute z_j^k following Algorithm 2 (y_j^k,β_D)
    Notify the sources with the new values of $z_{j,n}^k$
  else if $\rho_n \leq \rho_L$ **then**
    Relax flows: compute z_j^k following Algorithm 2 (y_j^k,β_I)
    Notify the sources with the new values of $z_{j,n}^k$
  Wait T_A seconds

4.1.2. Determining the adapted source rates of flows

Algorithm 2 is also implemented on the relay nodes. It computes the adapted source rates for each flow that will then be notified to the sources. Let $z_{j,n}^k$ denote the adapted source rates returned by node n for the flow originating at node j of class k. In our solution the values of $z_{j,n}^k$ strongly depend on the current assessment of the radio channel utilization. This algorithm is called by Algorithm 1 either to increase (with proportion β_I), or alternately, to decrease (with proportion β_D) the rates from the sources. Nonetheless, flows from different classes may undergo different adaptation as described below.

Initially, the adapted rate that will be assigned by node n to a flow from source j with priority k is set to 0, namely $z_{j,n}^k = 0$. The algorithm is composed of two iterative steps.

In the first step (lines $4-7$), the algorithm aims at allocating a certain portion of the available resources to each flow according to their current rate and class. Their respective rates are increased in a while loop with increments of length δ_k. For a given flow of class k, it stops increasing when a percentage, denoted by α_k, of the current rate has been reached, namely $z_{j,n}^k < \alpha_k \cdot y_j^k$.

In a second step (lines $8-11$), if the threshold on the overall transmission rate has not been reached ($\sum z_{j,n}^k < F \cdot \sum y_j^k$ where F is an input parameter of Algorithm 2 that sets the size of increments or decrements -β_I or β_D from Algorithm 1- for the overall transmission rate), the resources left are shared among the classes and sources with the same principle, i.e. increasing with increment δ_k.

Both thresholds α_k and δ_k depend on the flows class k. Typically, we have $\alpha^1 > \alpha^2 > \ldots > \alpha^K$ and $\delta^1 \geq \delta^2 \geq \ldots \geq \delta^K$. We discuss in further detail the parameter setting in the next section.

Algorithm 2: Run at each relay node n to determine the adapted source rates of flows.

Input: y_j^k, F

Output: $z_{j,n}^k$

1 $z_{j,n}^k = 0$
2 **while** $\sum_{j,k} z_{j,n}^k < F \cdot \sum_{j,k} y_j^k$ **do**
3 **for** (j,k) *in* $\{1,\ldots,S\} \times \{1,\ldots,K\}$ **do**
4 **if** $z_{j,n}^k < \alpha_k \cdot y_j^k$ **and** $\sum_{j,k} z_{j,n}^k < F \cdot \sum y_j^k$ **then**
5 $z_{j,n}^k = z_{j,n}^k + \delta^k$
6 **while** $\sum_{j,k} z_{j,n}^k < F \cdot \sum_{j,k} y_j^k$ **do**
7 **for** (j,k) *in* $\{1,\ldots,S\} \times \{1,\ldots,K\}$ **do**
8 **if** $\sum_{j,k} z_{j,n}^k < F \cdot \sum y_j^k$ **then**
9 $z_{j,n}^k = z_{j,n}^k + \delta^k$

4.1.3. Implementing the newly adapted flow rates at the sources

Algorithm 3 is implemented at the source nodes. For its flow of class k, the source node j may receive multiple values of $z^k_{j,n}$ where n refers to the relay node that computed the proposed flow rate. Every T_A seconds, Algorithm 3 readjusts the adapted rate for its flow of class k as the minimum of received value $z^k_{j,n}$. Denoting s^k_j as the adjusted rate of the flow of class k originating at node j, we have: $s^k_j = \min_n(z^k_{j,n})$ where n spans over the relay nodes that returned an adapted rate to node j. Note that in some case the nominal rate of a flow can be found to be less than its readjusted rate (i.e., $x^k_j < s^k_j$). Should it be the case, then the implemented flow rate would simply be the nominal one.

Algorithm 3: Run at each source node j to adjust the adapted flow rates.

Input: T_A, $z^k_{j,n}$
while *True* **do**
  Initialize all $z^k_{j,n} = \infty$
  Collect values $z^k_{j,n}$ from the relay nodes
  Compute $s^k_j = min_n(z^k_{j,n})$
  for k *in* $\{1, \ldots, K\}$ **do**
    if $y^k_j < s^k_j$ **and** $x^k_j \geq y^k_j$ **then**
      Increase y^k_j to its new value s^k_j
    if $y^k_j > s^k_j$ **then**
      Decrease y^k_j to its new value s^k_j
  Wait T_A seconds

5. Numerical results

We now describe the experimental setup to evaluate the performance of our proposed FDCC solution. First, we detail the simulation parameters and we explain how to select adequate values for our solution parameters. Next, we show the results obtained by FDCC for three different scenarios. The first one consists of a tree topology used to describe how FDCC works, while the two others represent more realistic topologies for smart grid NANs.

5.1. Simulation details

We implemented our solution in C++ within the ns-3 [16] simulator. The wireless communications between nodes are operated using the recent amendment 802.11ac of the IEEE 802.11 standard on a single 20MHz channel with a MCS (Modulation Coding Scheme) set to 0 so that transmissions occur at a physical rate of 6.5 Mbps. The maximum number of packets that can be buffered in every node has been set to 100 packets.

Source nodes deliver four different classes of traffic, typically corresponding to smart grid NAN applications with different traffic properties and QoS requirements. The different classes are presented in Table 2. As can be seen, different applications are grouped into each class. Therefore, a variable size (specifically, a truncated exponential distribution) has been chosen for the data packets. With respect to the inter-arrival packet time, the same criteria has been followed, and so packets are generated with different and random time between each other. Recall that class priorities are ranked in an ascending manner so that class 1 is viewed as the most critical. Nonetheless, let us remind that in our context, we consider that the traffic from all classes matters. It follows that our goal is not to entirely guarantee the best possible performance for class 1 traffic at the expense of the other classes but rather to be able to favor class 1 while keeping the communications from lower classes alive. All communications are performed using the transport protocol UDP.

Paths between source and destination nodes are found using the well-established Ad hoc On-Demand Distance Vector (AODV) routing protocol [17] wherein the weight of links derives from the expected transmission counts (ETX) to send a packet. To this end, the ETX implementation provided in [18] has been used.

Anyway, as it was mentioned in previous sections, our solution is agnostic to the routing protocol, and so other routing protocols can be used.

Each simulation corresponds to 100 seconds of network operation, and is run several times with different pseudo-random generator seeds. Besides, the first twenty seconds of each simulation are considered as a transient regime, and so results are not gathered during this time. Table 3 shows the main simulation parameters.

Our proposed FDCC solution involves several parameters that must be set before its execution. We briefly present guidelines on how to select them. K is simply determined by the number of traffic classes. T_A sets the frequency at which the solution results are updated. A value near 1 second seems to be good trade-off between reactiveness and bandwidth usage. β_D (resp. β_I) specifies the level at which sources decreases (resp. increases) the flow rates in case of over-utilization (resp. under-utilization). We select β_D =1.05 and β_I =0.75 in order to gradually coming closer the saturation threshold while quickly going away from it if reached. ρ_U and ρ_L defines the interval at which the radio channel is deemed adequately utilized (neither under-, nor over-utilized). In the case of links operating over IEEE 802.11, having an utilization factor between 0.7 and 0.8 provides high but not saturated utilization of the resources. The values of α^i indicate the relative importance of each traffic class with respect to the others. We recommend taking values in the range of 0.5 and 0.8. Finally, δ^i's values set the granularity of our solution and as so must be set small enough. In our case, we simply set them to 1. Table 4 summarizes the parameter values used for our congestion control mechanism.

For the performance evaluation, we compare FDCC with the EDCA mechanism [6]. In EDCA the traffic is classified in different categories according to their traffic classes. Each node has several internal queues with different parameters to prioritize traffic according to their categories. The EDCA default parameters, also used in our simulations, are shown in Table 5. The arbitration interframe space (AIFS) is the interval of time that the station must sense the medium as idle before trying to transmit the next frame. The transmission opportunity (TXOP) represents an allocation of time that can be used to transmit one or several frames in a sequence. Finally, for each category the minimum and maximum contention window (CW) are the limits from which the random backoff is uniformly chosen. Let us emphasize that the EDCA mechanism is local. It processes frames according only to their traffic classes and does not differentiate the flows from the different sources. Furthermore, EDCA mechanism is also independent of current state of the network (e.g., congestion).

Table 2: NAN applications transmitted over the smart grid.

Class	**Applications**	**Packet length average (Bytes) and PDF**	**Packet generation rate (per sec) average and PDF**
1	Demand Response, Outage Management	200 Exponential	100 Exponential
2	Video surveillance, Overhead Transmission Line Monitoring, Substation Automation systems (SASs)	200 Exponential	100 Exponential
3	Home Energy Managament (HEM), Electric Vehicles (EVs) Charging	200 Exponential	100 Exponential
4	Meter Data Management	200 Exponential	100 Exponential

Table 3: Main simulation parameters.

Description	Value
Network simulator.	ns-3.28
Simulation time	100 s
Tranport layer	UDP
Random number generator	MRG32k3a
Routing protocol	AODV
Routing metric	ETX
Maximum queue size	100 packets
Wireless physical layer	802.11ac
Modulation coding scheme	0
Short guard interval	1
Frame aggregation factor	0
Channel width	20 MHz

Table 4: Scenarios A and B - FDCC used parameters.

Parameter	Value
K	4
T_A	1 sec
β_D	0.75
β_I	1.05
ρ_U	0.8
ρ_L	0.7
$[\alpha^1, \alpha^2, \alpha^3, \alpha^4]$	$[0.8, 0.7, 0.6, 0.5]$
$[\delta^1, \delta^2, \delta^3, \delta^4]$	$[1, 1, 1, 1]$

Table 5: EDCA used parameters.

Class	CWmin	CWmax	AIFS	Max TXOP
4	15	1023	7	0
3	15	1023	3	0
2	7	15	2	3.008 ms
1	3	7	2	1.504 ms

5.2. Scenario A: Tree topology

In the first scenario, we consider a tree topology made of six nodes with three traffic sources (S1, S2 and S3), two relays (R1 and R2) and one destination (D) as depicted in Figure 2. As discussed above, each source node delivers traffic belonging to four different classes. In any case, packets are first sent to node R1, then forwarded to node R2 before finally reaching node D. The distances between nodes are chosen so that, together with the selected channel model, they lead to the network topology shown in the figure. More specifically, adjacent nodes are spaced by 80 meters with the exception of node pairs (S1-R1) and (S3-R3) that are 113 meters away. In all cases, adjacent nodes are always within each other carrier sensing range. Note that with a packet generation rate at each source node of 400 pkt/s (viz 0.64 Mbps), the combined workload of S1, S2 and S3 amounts to 1.92 Mbps and undoubtedly overflows node R1, whose transmission rate of 6.5 Mbps is reduced by half due to DCF overheads and furthermore shared with its four neighbors.

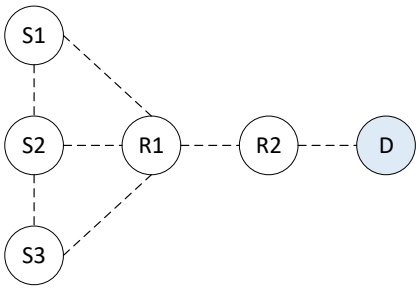


Figure 2: Scenario A - Tree topology scenario.

Next, the performance results obtained with our proposed mechanism (Fair Distributed Congestion Control, FDCC) are shown. These results are compared with those obtained without congestion control (Uncontrolled), and with those obtained when the Enhanced Distributed Coordination Funtion (EDCA) is activated. To begin with, we look at the channel utilization factor at the relay node R1. Channel utilization here refers to the occupation factor of the radio channel averaged over a period of time of 1 second. Figure 3 shows the corresponding results. The blue line (labelled *Uncontrolled*) represents the case where our solution does not operate and sources are free to generate packets at their initial rate, namely 100 pkt/s for each class. The channel utilization factor nears 0.8 (meaning that R1 perceives the radio channel as busy about 80% of the time). In practice, the radio channel can be viewed as fully saturated since the 20% left corresponds to unavoidable delays such as those incurred by the contention window. On the other hand, the red line (labelled *EDCA*) represents the case where our solution does not operate but the EDCA mechanism does. Here, each traffic class is assigned to an specific access category according to its priority (see Table 2). In this case, the channel utilization factor nears 0.84 (meaning that R1 perceives the radio channel as busy about 84% of the time). Finally, the green line corresponds to the case where our solution is applied (labelled *FDCC*). The ρ factor oscillates between 0.6 and 0.8 with a mean value of 0.71. This behavior indicates that, with our solution, channel saturations may occur but they are rapidly vanishing. On the other hand, Figure 4 shows the buffer occupancy in the same relay node R1. It can be seen that, when no mechanism is applied, the buffer is completely full most of the time, resulting in uncontrolled packet losses. In a similar way, when EDCA is activated, the buffer occupation is lower, but the buffer is also filled numerous times, and therefore packet losses also happen. Finally, when using FDCC, the controlled rate reduction makes the occupation of the buffer much lower, not reaching the total occupancy at any time and thus avoiding packet losses. In fact, the mean number of packets in the buffer for the Uncontrolled, EDCA and FDCC cases are 93.55, 19.88 and 9.56, respectively.

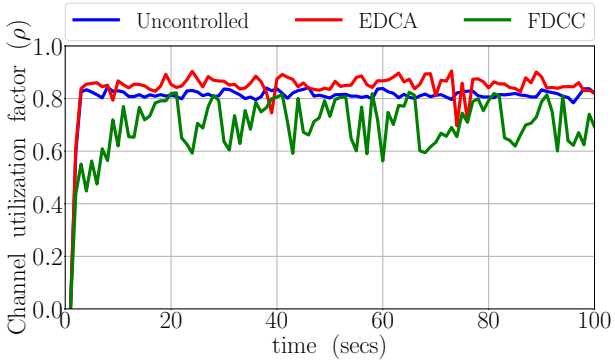


Figure 3: Evolution of the channel utilization factor seen by node R1.

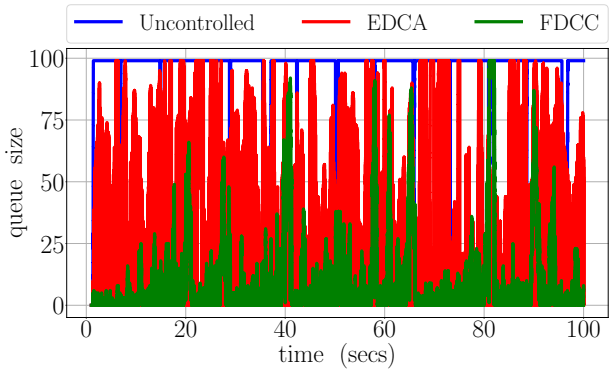


Figure 4: Evolution of the buffer occupation at node R1.

We now study the packet generation rates at the source node S1. Figure 5 represents, for each traffic class, the minimum and maximum values, together with the percentile boxes (25, 50 and 75-percentiles). As expected, in the absence of a generation rate control mechanism, this rate holds its initial value of 100 pkt/s for any class. On the other hand, with the help of our solution, we can observe that the classes exhibit different patterns. For instance, the class 1 boxplot indicates that the generation rate may vary from 75 up to 95 pkt/s while its median value is approximately 80 pkt/s. Looking at the less critical classes, we note that the their packet generation rate tends to be significantly smaller but yet far from 0 though.

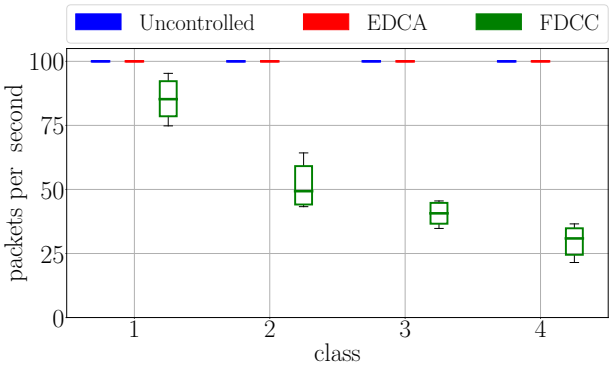
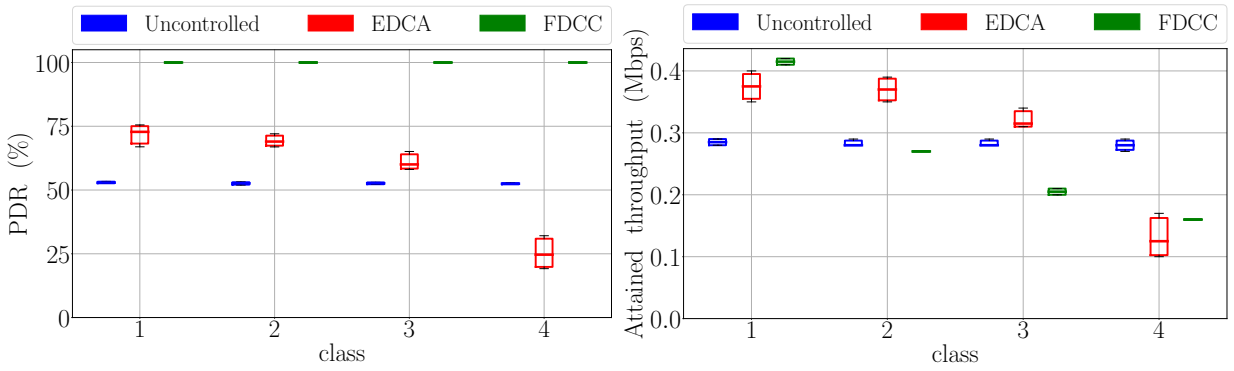


Figure 5: Packet generation rate at node S1 for each traffic class.

In Figure 6a, we study the packet delivery ratio (PDR) attained for each class. Because the radio channel tends to be saturated in the absence of our solution (see Figure 3), the corresponding PDR for class 1 only peaks approximately at 50%. Besides, as no distinction is made between the different traffic classes, the values is the same for all of them. This fact does not occur when EDCA is applied. As can be seen in the figure, the PDR value decrease from 75% to 25% depending on the traffic class. Finally, with FDCC, the fact of having a previous generation rate control results in a PDR of 100%.

Combining the two previous performance parameters, generation rate and PDR, we can obtain the attained throughput by the different applications depending on the class they belong to, which is shown in Figure 6b. This parameter refers to the number of bytes that were successfully transmitted from the sources to the destination node. Not surprisingly, we notice that in the absence of any control mechanism, all classes exhibit the same pattern with a constant throughput equal approximately to 0.28 Mbps. On the other hand, both EDCA and FDCC mechanisms cause different attained throughput depending on the traffic class. As will be shown with the next evaluation scenario, another advantage of FDCC is a better distribution of this throughput between all the network nodes.



(a) PDR. (b) Attained throughput.

Figure 6: Packet delivery ratio (PDR) and attained throughput at node D for each traffic class.

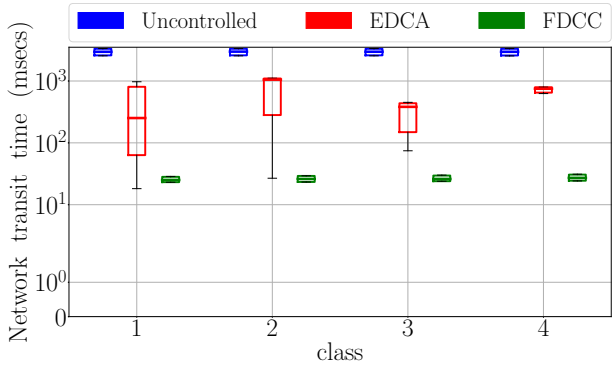


Figure 7: Network transit time for each traffic class.

Finally, the network transit time values are shown in Figure 7. Again, when no additional mechanisms are applied, the value is the same for all traffic classes. As can be seen, both EDCA and FDCC make the values lower and dependent on the traffic class. Besides, FDCC achieves significant improvements in this parameter for all traffic classes. Interestingly, unlike EDCA, this decrease with FDCC is obtained while maintaining the same proportion of traffic from the farthest nodes to the gateway, as will be shown again with the next scenario.

As a final experiment in this simple scenario, we show the ability of the FDCC mechanism to regulate the rate of all source nodes at the same value and for each traffic class. In this case, the generation rate of the sources S1, S2 and S3 has been configured, for each traffic class, to 300, 200 and 100 pkt/s respectively. Figure 8 shows the results obtained once the FDCC mechanism has performed its rate control. As can be seen, for traffic class 1, the rate of sources S1 and S2 has been reduced to the same value of approximately 150 packets per second since the network does not have enough resources to serve the highest quantities requested. However, the generation rate of source S3 has not been altered, since that source requested a lower value which is possible to attain. For the rest of the traffic classes, the generation rate of all the nodes has been reduced, adjusting them to very similar values.

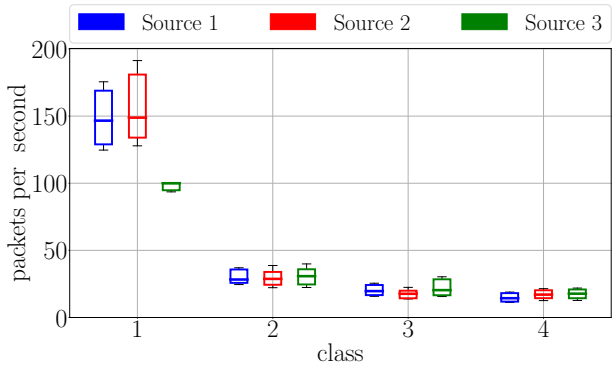


Figure 8: FDCC resulting generation rates for source nodes with unbalanced packet generation rates.

Overall, this first scenario explains the way FDCA works, and illustrates its usefulness. In the next section, we consider a second scenario where the number of nodes (sources and relays) is much larger, and make a more detailed evaluation.

5.3. Scenario B: Smart grid scenario

The second scenario deals with a network where nodes are arranged in a square grid. We alternately study the case of a 9-node and 25-node grids as illustrated in Figure 9. In both cases, the distance between adjacent nodes is set to 100 meters so that nodes can only communicate with their horizontal and vertical neighbors (if any). In this scenario, one node serves as destination (data concentrator) while other nodes act as traffic sources and intermediate routers. Source nodes generate data traffic belonging to the four traffic classes, and directed to the data concentrator, in the same way than in the previous scenario (see Table 2). Clearly, given the workload delivered by each node, the network will suffer congestion unless some form of control is made on the sources packet generation rate.

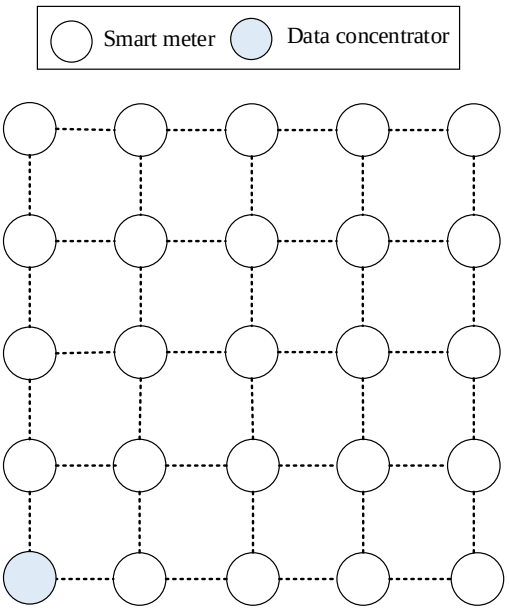
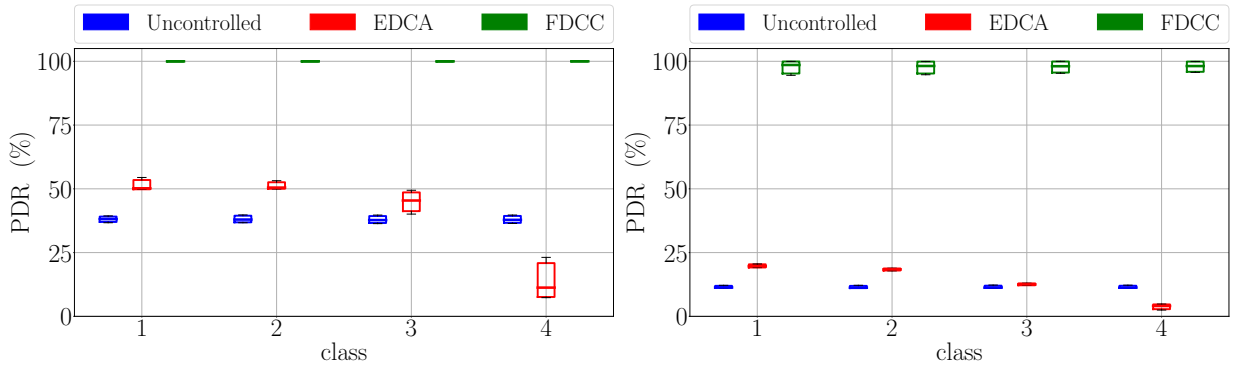


Figure 9: Scenario B - Smart grid Neighborhood Area Network scenario for 25 nodes.

First, we study the packet delivery ratio for each traffic class. Two different network sizes have been considered,

one of them with 9 network nodes and the other with 25. Figure 10 shows the PDR results for the two cases. As can be seen, in the absence of our solution a high percentage of packets will be lost on their way towards the data concentrator. The loss of packets is greater when the size of the network grows. Remember that all nodes try to transmit equally, and therefore as the number of nodes grows, the attempt to use wireless channels also grows, leading to a higher congestion situation. Note that packets that are lost along their way to the data concentrator have previously consumed network resources in a completely useless way. All these disadvantages can be resolved with a smart generation rate control at the source nodes, achieving a PDR near 100% as can be seen in the figure when the FDCC mechanism is applied.

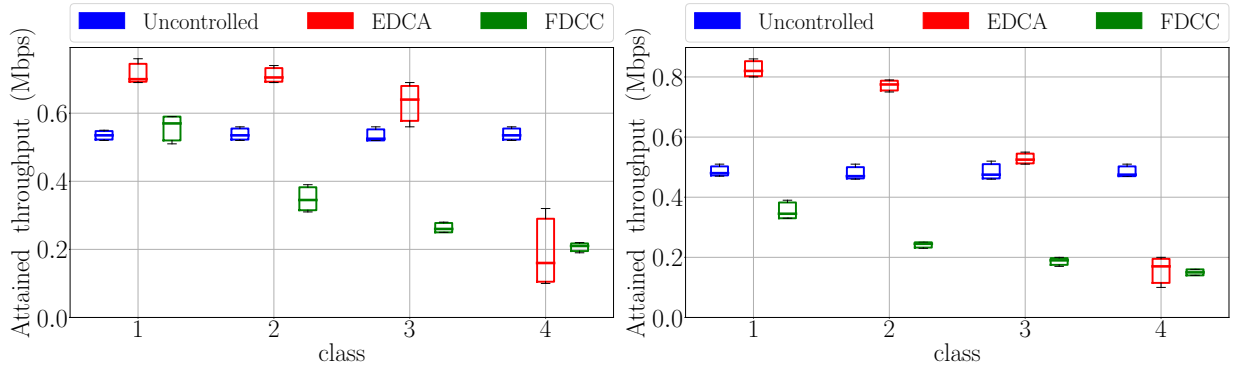


(a) Network size 9 nodes

(b) Network size 25 nodes.

Figure 10: Packet delivery ratio (PDR) for each traffic class and for different network sizes.

To have a better understanding on the network performance, we look now at the attained throughput. Figure 11 shows the results for the two network sizes under consideration. Again, it can be seen that the attained throughput by the different traffic classes is the same when differentiation mechanisms are not applied. EDCA does provide differentiation, allocating more network resources to the most priority traffic, as does FDCC. Given the set of parameters considered for these two mechanisms in these simulations, we observe that the overall throughput attained by EDCA exceeds that obtained by FDCC. In fact, the attained throughput obtained with FDCC is even lower than the obtained without applying any mechanism. These values can of course be modified by setting less restrictive parameters for FDCC. However, and more importantly, the attained throughput with FDCC has a higher quality in two ways as will be seen below: fairness in the resource sharing among the different network nodes, and network transit time.



(a) Network size 9 nodes.

(b) Network size 25 nodes.

Figure 11: Attained throughput for each traffic class and for different network sizes.

So far, the combination of Figures 10 and 11 has shown that applying our solution enables all applications to experience a near 100% packet delivery ratio and to differentiate their attained throughput based on their assumed

criticality while maintaining a high level of the network resource utilization. As previously said, next we are going to study the fairness in the network resources distribution among the different source nodes. The point here is to observe how, in the absence of a fair mechanism, the nodes that are closer to the destination obtain a better service from the network (higher throughput) than those that are further away. This fact is clearly seen in Figure 12, which shows the attained throughput obtained for each traffic class, depending on the distance from the source node to the destination (measured in number of hops). The network size considered here is 25 nodes, and as it can be seen in the figure the distance from the source nodes to the destination varies between 1 and 8 hops. With both uncontrolled and EDCA modes, and for all traffic classes, nodes located at a distance greater than four hops get very poor performance. In addition, as can be seen with the boxplots, the variability presented by the results is very high. However, applying the FDCC mechanism, the value of the flow rate is practically constant with regard to the number of hops, and with a very small variability in the results. Both facts are extremely important, since on the one hand the same service is offered to all users, and on the other the quality of the experience of each of them remains constant over time. This fact is especially serious for traffic classes 3 and 4, which would be otherwise virtually eliminated when they originate from nodes located more than four hops away.

To complement and quantify these results, in Figure 13 we present the value of the Jain's index calculated on the attained throughput by traffic flows coming from different nodes. As discussed in Section 3, the Jain's index is a common measure for fairness [15]. We observe that the value of this index is considerably higher when the FDCC mechanism is applied, with values very close to the maximum. On the other hand, when the size of the network grows, as expected the variability between nodes is even greater, so that the improvements obtained with FDCC are even more relevant.

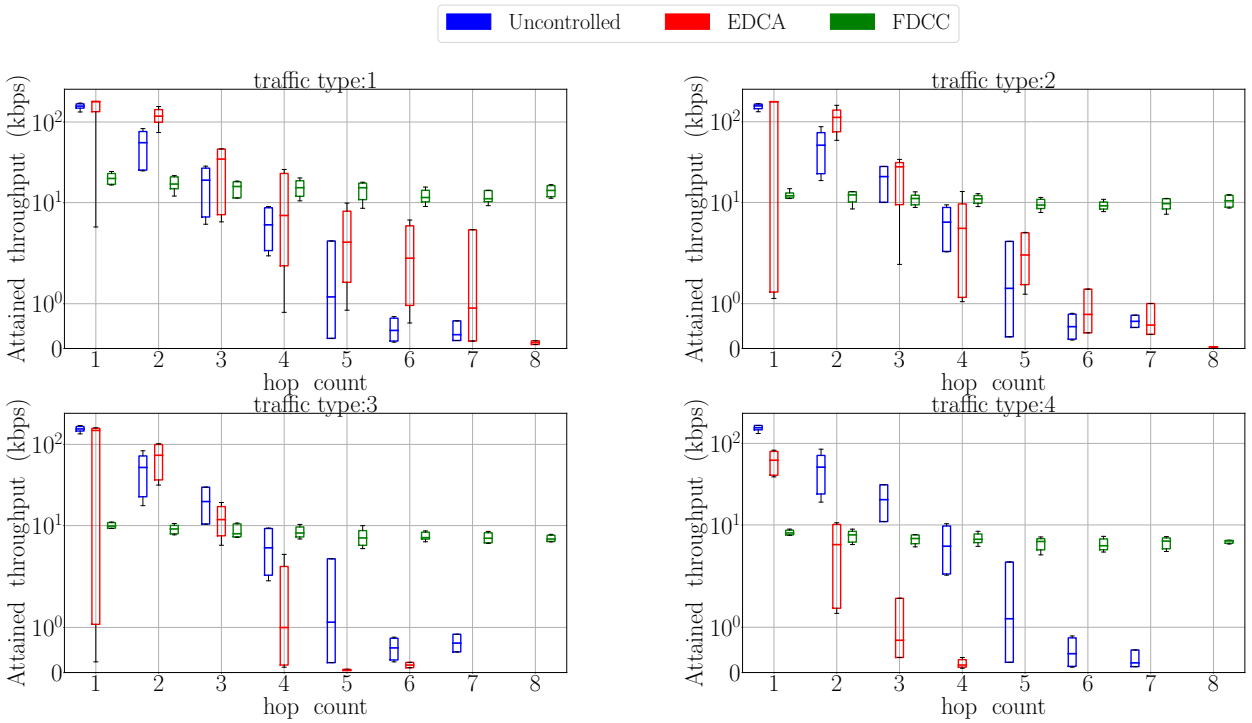
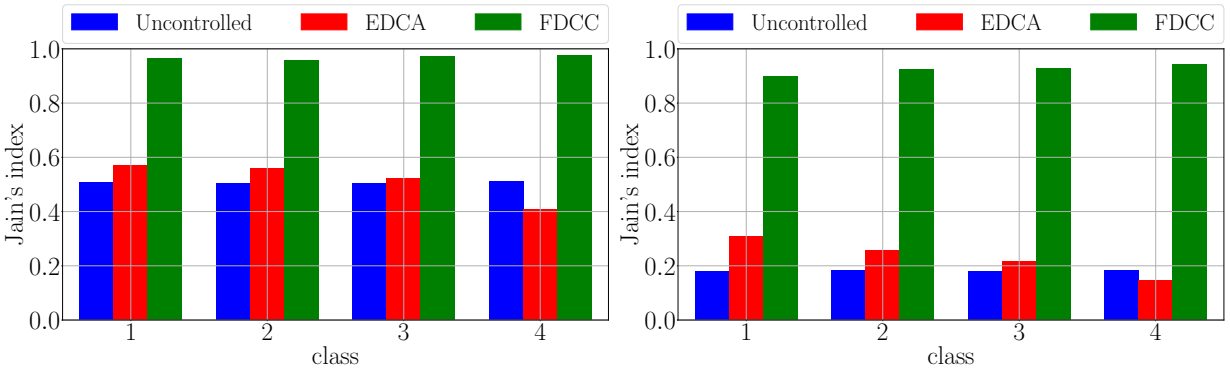


Figure 12: Throughput vs number of hops for each traffic class (network size 25 nodes).

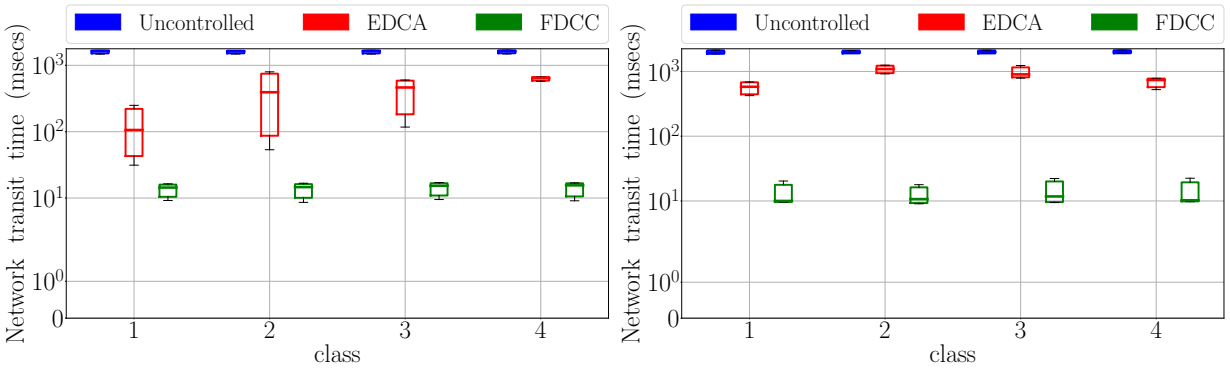


(a) Network size 9 nodes. (b) Network size 25 nodes.

Figure 13: Fairness on the attained throughput for each class and for different network sizes.

In addition to the avoidance of network congestions and buffers overflows, implementing a packet generation rate control typically leads to lower packet network transit times. This is confirmed by Figure 14 where both EDCA and FDCC manage to reduce this time for all traffic classes, with a more significant decrease in the case of FDCC. Note also that in both uncontrolled and EDCA modes, packets from distant nodes are dropped halfway to their destination. Therefore their transit times, which would be typically large if they had arrived to their destination, are not taken into account when calculating the average value of the network transit time. However, with FDCC these traffics are regulated so that the obtained average values for the network transit time are still significantly lower.

Regarding the network transit time, an interesting measure is the percentage of packets of each class that has been delivered to their destination in a time less than a given maximum delay. Clearly this maximum delay depends on the importance (class) of each traffic flow. In our case, we use the values given in Table 6. We refer to the percentage of packets meeting their time constraint as the compliant factor and we represent its values in Figure 15. In addition to the factor for each class, a global value is offered, taking into account the proportion of packets of each class. Our results indicate that the value of this factor is considerably better in all cases when applying FDCC, taking practically its maximum value.

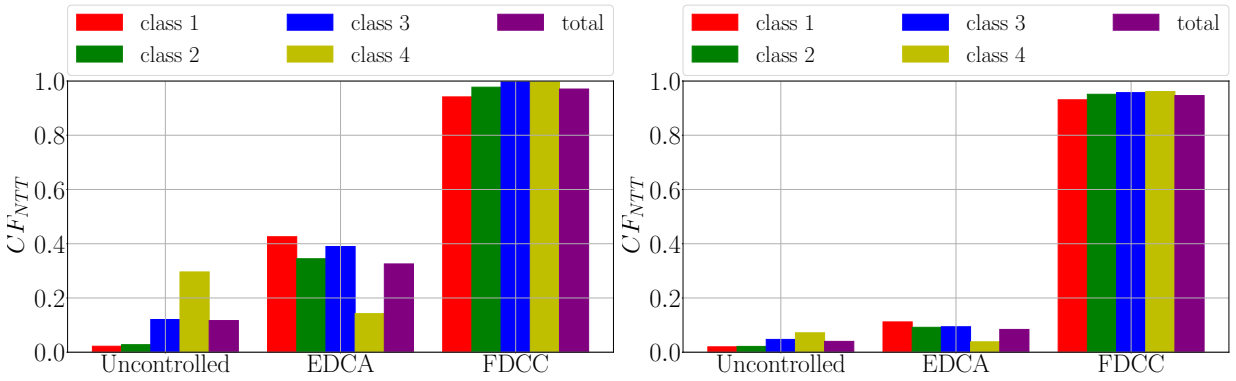


(a) Network size 9 nodes. (b) Network size 25 nodes.

Figure 14: Network transit time for each class and for different network sizes.

Table 6: Maximum allowed network transit times.

Class	Network transit time
1	50 ms
2	100 ms
3	1000 ms
4	2000 ms



(a) Network size 9 nodes.

(b) Network size 25 nodes.

Figure 15: Network transit time compliant factor.

Finally, we have measured the additional control traffic introduced by the implementation of FDCC. This traffic corresponds to the packets generated by the routers to indicate to the traffic sources the generation rate that they are allowed to transmit for each category. The amount of traffic generated depends on the averaging period selected for the algorithm (T_A). But it must also be taken into account that control packets are only generated when new allowable rate values (that is, values that will modify the current generation rate) are obtained by the routers. Specifically, for all the experiments carried out, the value of this control traffic is always between 3 and 4% of the data traffic.

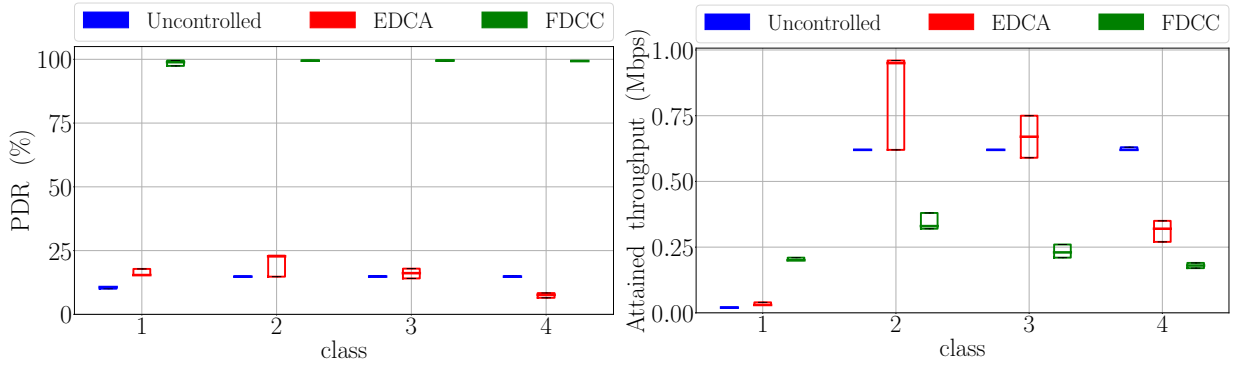
5.4. Scenario C: Downstream traffic from the control center to the NAN

As a third simulation scenario, we now consider the inclusion of downstream data traffic from the control center destined to each of the network nodes. Within the scope of the NAN, this traffic will therefore have the data concentrator as the source node. In order to demonstrate the versatility of the FDCC mechanism under various configurations, let us consider that the downstream traffic coming from the control center is of extreme importance, included in traffic class 1, so that it will not be affected by the rate control mechanism. In other words, the data concentrator has to forward all the received packets from the control center. The rest of the network nodes will generate traffic belonging to classes 2, 3 and 4. Note that the rate control mechanism applies to these traffic classes as in the previous scenarios. Table 7 shows the set of parameters chosen for the new downstream traffic. Its packet generation rate depends on the number of network nodes (here 25), and has been set to 5 pkt/s per destination node. The FDCC mechanism selected parameters are similar to those presented in Table 4, the only difference being in the α values assigned to each class. Given that the packet generation rate of NAN applications that belong to class 1 is not controlled, the α values for this class is not needed. For classes 2, 3 and 4, the chosen values are 0.8, 0.7 and 0.6 respectively.

Table 7: Downstream data traffic.

Class	Applications	Packet length average (Bytes) and PDF	Packet generation rate (per sec) average and PDF
1	Demand Response, Outage Management	200 Exponential	5 Exponential

In this scenario, we study a network arranged in a grid with 25 nodes (see Figure 9) and we evaluate its performance in terms of PDR, attained throughput, network throughput fairness and transit time. Figure 16 shows the PDR and the attained throughput. Similarly to the previous scenarios, Figure 16a shows how the PDR reaches almost 100% with our solution compared to the other two cases where the PDR is less than 25% for all the traffic classes. Regarding the attained throughput, we notice in Figure 16b that the value for traffic class 1 is significantly higher when FDCC is applied. In fact, the obtained value corresponds to all the throughput demanded by the data concentrator (since the generation rate of this traffic is not regulated and the PDR is 100%). For the rest of the traffic classes, the FDCC mechanism tends to act likewise to the previous scenarios.



(a) PDR.

(b) Attained throughput.

Figure 16: Packet delivery ratio (PDR) and attained throughput for each class (network size 25 nodes).

The attained throughput fairness between the network nodes can be seen in Figure 17, where the Jain's index values are shown for each traffic class. Not surprisingly, for traffic class 1, the Jain's index has always the maximum value, since this traffic is only sent by the data concentrator. For the rest of the traffic classes, the distribution of the attained throughput between the different nodes is much more uniform when FDCC is working.

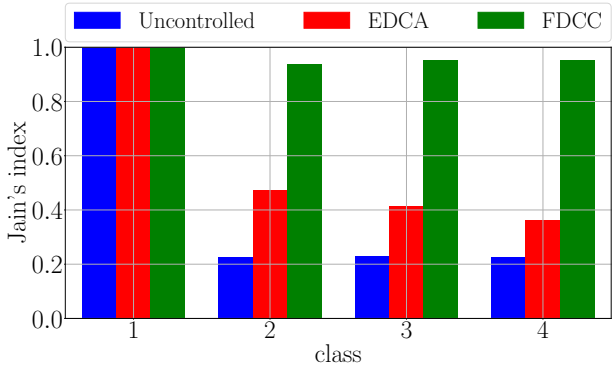
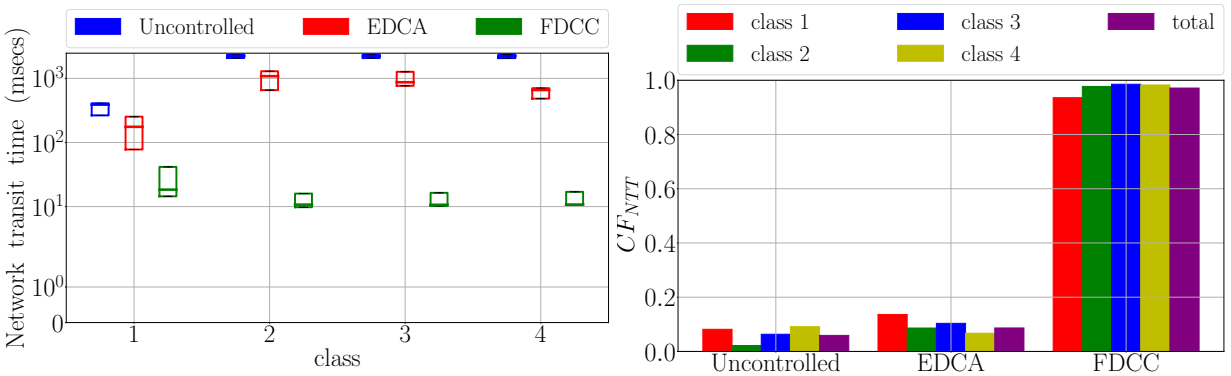


Figure 17: Fairness on the attained throughputs for each class (network size 25 nodes).

Finally, Figure 18 shows how in this scenario the improvements obtained in terms of packet network transit time are also achieved. In Figure 18a it can be seen how the values are lower with FDCC for all the traffic classes, including the new and uncontrolled downstream class 1 traffic. Regarding the compliant factor, the obtained values are depicted in 18b, where the maximum values selected in Table 6 have been taken into account. As it can be seen, the minimum requirements for the network transit times are significantly better accomplished for all the traffic classes when our solution is implemented.



(a) Network transit time.

(b) Compliant factor.

Figure 18: Network transit time and compliant factor for each class (network size 25 nodes).

6. Conclusions

In this paper, we have presented a conceptually simple congestion control mechanism for an IEEE 802.11-based NAN that carries traffic from several classes with different QoS requirements. NANs are typically multi-hop networks so that handling priorities only at a link-level would undoubtedly lead to favoring nodes close to the data concentrator (NAN gateway). Our solution relies on three distributed algorithms executed on the smart meters that work together to adequately regulate the traffic sources with regard to their class and hence avoiding network overload.

Our solution accounts for different traffic classes (with different QoS) while ensuring a fair sharing of the network resources between them and, importantly, and granting access to the network resources even for sources far from the data concentrator (with regards to the number of hops).

Using a discrete-event simulator, we have run a number of experiments to evaluate the performance of our solution and compared its results with those obtained by EDCA that works only at the link-level. Our numerical results show that our proposed solution is efficient in terms of packet delivery ratio and transit time. While its performance at network throughput may appear a bit lower than EDCA, this is only the result of a better fairness in the sharing of network resources.

Going further in the search for better performance of congestion control mechanisms in multi-hop wireless networks, our work is currently branched off into several lines of research. On the one hand, the applicability of the algorithms proposed in this article in other communications network environments will be studied, such as in crowded networks with mobile nodes. These networks can be built for example through the intercommunication of cell phones of people who roam their cities. Here, all the network nodes (or a selected group among them) could periodically broadcast the allowable maximum generation rate as a function of the measured channel utilization factor and the number of 1-hop and 2-hops neighbors. On the other hand, we are also applying different machine learning algorithms and evaluating their performance for congestion control on NANs.

Acknowledgments

This work was supported by the Spanish Research Council under project MAGOS (TEC2017-84197-C4-3-R). Juan Pablo Astudillo León is the recipient of a full scholarship from the Secretaría de Educación Superior, Ciencia, Tecnología e Innovación (SENESCYT), Ecuador.

The authors wish to thank the anonymous referees for their thorough and constructive review of an earlier version of this paper.

References

[1] N. Shaukat, S. Ali, C. Mehmood, B. Khan, M. Jawad, U. Farid, Z. Ullah, S. Anwar, M. Majid, A survey on consumers empowerment, communication technologies, and renewable generation penetration within smart grid, Renewable and Sustainable Energy Reviews.

[2] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, G. P. Hancke, A Survey on Smart Grid Potential Applications and Communication Requirements, IEEE Transactions on Industrial Informatics 9 (1) (2013) 28–42. arXiv:arXiv:1011.1669v3, doi:10.1109/TII.2012.2218253.
URL http://ieeexplore.ieee.org/document/6298960/

[3] W. Meng, R. Ma, H. H. Chen, Smart grid neighborhood area networks: A survey, IEEE Network 28 (1) (2014) 24–32. doi:10.1109/MNET.2014.6724103.
URL http://ieeexplore.ieee.org/document/6724103/

[4] Y. Tsado, K. A. A. Gamage, D. Lund, B. Adebisi, Performance analysis of variable Smart Grid traffic over ad hoc Wireless Mesh Networks, in: 2016 International Conference on Smart Systems and Technologies (SST), IEEE, 2016, pp. 81–86. doi:10.1109/SST.2016.7765637.
URL http://ieeexplore.ieee.org/document/7765637/

[5] T. Clausen, P. Jacquet, Optimized Link State Routing Protocol (OLSR), Internet Engineering Task Force (IETF) 4 (2003) 75. arXiv:arXiv:1011.1669v3, doi:10.1.1.11.620.
URL https://www.ietf.org/rfc/rfc3626.txt

[6] I. S. for Information technology Telecommunications, information exchange between systems Local, metropolitan area networks Specific requirements Part 11: Wireless LAN Medium Access Control (MAC), P. L. P. Specifications, IEEE Std 802.11-2016 (revision of ieee std 802.11-2012), pp. 1 – 3534, IEEE Computer Society.

[7] Y. Tsado, K. Gamage, B. Adebisi, D. Lund, K. Rabie, A. Ikpehai, Improving the Reliability of Optimised Link State Routing in a Smart Grid Neighbour Area Network based Wireless Mesh Network Using Multiple Metrics, Energies 10 (12) (2017) 287. doi:10.3390/en10030287.
URL http://www.mdpi.com/1996-1073/10/3/287

[8] H. Gharavi, B. Hu, Multigate Communication Network for Smart Grid, Proceedings of the IEEE 99 (6) (2011) 1028–1045. doi:10.1109/JPROC.2011.2123851.
URL http://ieeexplore.ieee.org/document/5768102/

[9] X. Deng, L. He, X. Li, Q. Liu, L. Cai, Z. Chen, A reliable QoS-aware routing scheme for neighbor area network in smart grid, Peer-to-Peer Networking and Applications 9 (4) (2016) 616–627. doi:10.1007/s12083-015-0331-5.
URL http://link.springer.com/10.1007/s12083-015-0331-5

[10] J.-S. Jung, K.-W. Lim, J.-B. Kim, Y.-B. Ko, Y. Kim, S.-Y. Lee, Improving IEEE 802.11s Wireless Mesh Networks for Reliable Routing in the Smart Grid Infrastructure, in: 2011 IEEE International Conference on Communications Workshops (ICC), IEEE, 2011, pp. 1–5. doi:10.1109/iccw.2011.5963578.
URL http://ieeexplore.ieee.org/document/5963578/

[11] J. Kim, D. Kim, K.-W. Lim, Y.-B. Ko, S.-Y. Lee, Improving the reliability of IEEE 802.11s based wireless mesh networks for smart grid systems, Journal of Communications and Networks 14 (6) (2012) 629–639. doi:10.1109/JCN.2012.00029.
URL http://ieeexplore.ieee.org/document/6412861/

[12] J. P. Astudillo León, L. J. De la Cruz Llopis, A joint multi-path and multi-channel protocol for traffic routing in smart grid neighborhood area networks, Sensors 18 (11). doi:10.3390/s18114052.
URL https://www.mdpi.com/1424-8220/18/11/4052

[13] J. P. Astudillo León, L. J. de la Cruz Llopis, Emergency aware congestion control for smart grid neighborhood area networks, Ad Hoc Networks 93 (2019) 101898. doi:https://doi.org/10.1016/j.adhoc.2019.101898.
URL http://www.sciencedirect.com/science/article/pii/S1570870519303567

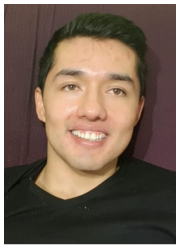
[14] J. P. Astudillo León, T. Begin, A. Busson, L. J. de la Cruz Llopis, Towards a distributed congestion control mechanism for smart grid neighborhood area networks, in: Proceedings of the 16th ACM International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks, 2019, pp. 29–36.

[15] R. Jain, D. M. Chiu, W. R. Hawe, A quantitative measure of fairnessand discrimination for resource allocation in shared computer system, Research Report DEC-TR-301, Digital Equipment Corporation, 1984.

[16] N. Consortium, Ns-3 a Discrete-Event Network Simulator (2019).
URL https://www.nsnam.org/

[17] C. Perkins, E. Belding-Royer, S. Das, Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561 (Experimental), Tech. Rep. 3561 (2003).
URL http://www.ietf.org/rfc/rfc3561.txt

[18] N. J. Jevtić, M. Z. Malnar, Implementation of etx metric within the aodv protocol in the ns-3 simulator, Telfor Journal 10 (1) (2018) 20–25.



Juan Pablo Astudillo León received the electronics engineering degree from the Universidad Politécnica Salesiana (Cuenca, Ecuador) in 2012, and the specialist in telecommunications technologies and the master in telecommunications engineering degrees from the Universidad de Buenos Aires (Buenos Aires, Argentina) in 2014 and 2015 respectively. He was assistant professor at the Universidad Politéctnica Salesiana in 2016. He is currently a Ph.D candidate in the SISCOM (Smart Services for Information Systems and Communication Networks) research group at the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain. His research includes wireless mesh networks, wireless mobile ad-hoc networtks, smart grids, traffic engineering and artificial intelligence .



Thomas Begin holds an Ingénieur degree in Telecommunication from ISEP (Paris) in 2003, and a M.Sc. and a Ph.D. degrees in Computer Science from UPMC (Univ. Paris 6) in 2005 and 2008, respectively. He was a post-doctoral fellow at UC Santa Cruz in 2009. Since 2009, he is Associate Professor at UCBL (Univ. Lyon 1) in the Computer Science department. During the academic year 2015-2016, he was on a research leave at the Univ. of Ottawa. He obtained the Habilitation Degree in Computer Science at UCBL (Univ. Lyon 1) in 2018. His research interests are in performance evaluation, computer network, and system modeling. His principal applications pertain to high-level modeling, wireless networks, resource allocation and queueing systems.



Anthony Busson has been full professor at University Claude Bernard Lyon 1 and LIP (Laboratoire de linformatique du paralllisme) since 2012 in Lyon (France). He works for 15 years on wireless networks: ad-hoc and infrastructure based networks. He is interested in mobility management, resource planning, and optimal configuration for different types of networks: Wi-Fi infrastructure based networks, vehicular ad-hoc networks, wireless mesh networks, and mobile ad-hoc networks. Solutions are evaluated through simulations, and mathematical tools (generally stochastic ones).



Luis J. de la Cruz Llopis received the telecommunication engineering degree in 1994 and the Ph.D. in telecommunications engineering in 1999, both from the Universitat Politécnica de Catalunya (UPC), Barcelona, Spain. He is currently an Associate Professor at the Department of Network Engineering of the UPC. His current research interests include wireless networks and IoT smart services.