SBR: A Novel Architecture of Software Defined Network Using the RPL Protocol for Internet of Things

PAUL SANMARTIN¹,², (Member, IEEE), KAREN AVILA², SEBASTIAN VALLE², JAVIER GOMEZ³, AND DALADIER JABBA², (Senior Member, IEEE)
¹System and Computer Engineering Department, Universidad Simón Bolívar, Barranquilla 080002, Colombia
²System and Computer Engineering Department, Universidad del Norte, Barranquilla 081007, Colombia
³Department of Telecommunication Engineering, Universidad Nacional Autónoma de México, Coyoacan, Ciudad de México D.F. 04510, Mexico

Corresponding author: Paul Sanmartin (psanmartin@unisimonbolivar.edu.co)

This work was supported by Universidad del Norte and Universidad Simón Bolívar.

ABSTRACT This paper proposes a novel architecture named SBR, a software defined network (SDN) architecture using as a routing protocol RPL, a Protocol for Low-Power, and Lossy Networks for Internet of Things (IoT). The architecture includes the function SIGMA, an objective function used in IoT applications to take routing decisions. This research represents an effort to relate IoT and SDN. This association was developed using RPL-SIGMA to solve various IoT and Wireless Sensor Networks (WSN) problems. At the end, a comparison between SBR (SDN-RPL with SIGMA) and SDN-RPL (without SIGMA) was included. The simulations show that SBR has a better performance concerning packet loss, latency, and energy consumption over an architecture based on IoT without SIGMA. The simulation also demonstrates that SDN can be used to manage the IoT network efficiently.

INDEX TERMS QoS, routing protocol, IoT, SDN, RPL, WSN.

I. INTRODUCTION

Internet of Things (IoT) is a concept that has generated a positive impact in the world with applications in industry, academia, and research, among others. According to ITU [1] (International Telecommunication Union), the IoT is a “global infrastructure of the information society, which offers advanced services interconnecting things,” having a predominant role due to they are increasing the connectivity of these objects. In other words, the interconnection of these devices (objects) belongs to the IoT networks or IoT domains, and they can be either of public or private use [2]. In the IoT context, these devices receive the name of routing nodes.

As it is known, routing allows the efficient exchange of information among IoT objects. However, routing in IoT inherits the problems of ad-hoc networks and sensor networks. These are part of the called IoT domains (networks), keeping in mind the characteristics of the devices that make up these networks.

The associate editor coordinating the review of this manuscript and approving it for publication was Thanh Ngoc Dinh.

The above leads to sharing some challenges such as limited resources [3], [4], routing in dynamic topology, and scalability. Nevertheless, other drawbacks arise when taking into account the priority of the packets in the low power and lossy networks (LLN) power consumption and packet loss [5].

Plenty of manufacturers, institutes, and universities have begun to address routing in IoT [6] through research and the use of different paradigms and technologies such as SDN. SDN is considerably growth and accepted globally, although several countries are still wary of migrating towards this technology. SDN is well recognized to manage all Wireless Sensor Network (WSN) aspects, even though its implementation is not widely deployed. However, from the academy, it has given excellent results.

Commonly SDN is used in wired networks [7]. Recently some researchers like [8] and [9] explain how OpenFlow controllers can increase the performance of wireless networks, Wireless Sensor Network (WSN), Wireless Mesh Network (WMN), and cellular networks [10]. As the first step in this research, an effort to associate IoT and SDN is achieved by articulating SDN with RPL as its routing protocol.
The proposal then addresses the concepts of routing in wireless networks, specifically in LLN, by defining a novel architecture of SDN using the RPL protocol and the SIGMA objective function to increase the quality of service in IoT (the proposed novel architecture is named SBR). The protocol used in this proposal is called IPv6 Routing Protocol for Low Power and Lossy Network (RPL) [11], [12], and the IETF standardized it under ROLL review [13], [14].

RPL is a source routing protocol that has been designed to operate on LLN networks [15], and it uses the IEEE 802.15.4 PHY by default. Several versions of LLN [16] and RPL networks have been developed to improve QoS management, such as in [17] and [18]. In RPL for the topology, the Destination-Oriented Directed Acyclic Graph (DODAG) concept is used [19], [20]. DODAG defines various types of control messages such as DODAG Information Object (DIO), DODAG Information Solicitation (DIS), Destination Advertisement Object (DAO), and Destination Advertisement Object Acknowledgement (DAO-ACK) [21].

The contributions of this work are the following: First, an architecture based on SDN and including RPL as its routing protocol has been implemented in ContikiOS. Second, researchers in the same area of interest can implement their proposals and compare them with this research. Third, future implementations will provide new alternatives of solution using SDN with RPL and SIGMA. Another contribution is that a novel architecture of SDN using the RPL protocol and SIGMA (SBR) has been proposed to manage the IoT networks. Finally, an SDN architecture using a cluster topology has been implemented to administer better and control the network.

At the end of this research, a comparison between SBR and SDN-RPL (without SIGMA) was developed to demonstrate the advantage of including SIGMA as the objective function in its architecture. The simulations show that SBR has a better performance concerning packet loss, latency, and energy consumption over an architecture based on IoT but without SIGMA.

The rest of the paper is divided into 7 sections: Section 2 is about the concept of SDN, in section 3, related works are explained, section 4 defines the proposed SBR architecture, section 5 describes the design and simulation of the proposed network, section 6 presents the results, and finally, section 7 presents some concluding remarks. surrounding equation.

II. SDN CONCEPT
Software Defined Networking (SDN) [22], [23] is taken into account for routing protocols in IoT, applying quality of service according to the needs of the network. Different approaches based on SDN have been developed, such as SDN for Wireless networks [24] and different IoT services [25]. It seeks to dynamically meet the differentiated levels of quality for the different tasks of the IoT in every different wireless network scenario [26].

A. SDN ARCHITECTURE
According to the Open Networking Foundation (ONF), SDN is an emerging network architecture where network control is decoupled from forwarding and is directly programmable [27], [28] Figure 1 depicts a view of a general SDN architecture. As shown in the picture, the SDN architecture separates the network control functions from the data forwarding plane [29]. The network control functions are centralized in one or several controllers [30].

OpenFlow (OF) is defined as a communication protocol that most SDN controllers use it. The OF protocol enables the control of network devices and the communication between SDN controllers and network nodes. The following are some other important protocols: Opflex [31], LISP [32], NetConf (Network Configuration Protocol) [33], IEEE P1520 [34], ForCES (Forwarding and Control Element Separation) [35], and SoftRouter [36].

B. SDN IN WIRELESS NETWORKS
SDN is used in wired networks [7]. However, recently some research [9] explains how OpenFlow controllers can increase the performance in wireless networks [8] and WSN, WMN, and cellular networks. Additionally, WSN and WMN can be benefited from SDN considering their multi-hop scenarios, contrary to WLANs and cellular networks that frequently show topology changes. Figure 2 displays a software defined wireless sensor network architecture based in Tanzeena et al. [37], describing data and control planes.
Advantages and disadvantages are:

- Advantage: The control plane consists of one (or possibly more) controller that centralizes all the network intelligence, performing network control such as routing and QoS control.
- Disadvantage: The architecture has never been tested neither in a simulation nor in a real network.

Costanzo et al. [38] present a similar proposal to analyze the opportunities and challenges of applying the SDN in WSN. Some advantages and disadvantages are:

- Advantage: It is based on the OpenFlow protocol and does not explain how to apply QoS.
- Disadvantage: it has never been tested in simulation scenarios.

Rahmani et al. [40] investigate the network performance of logical clustering applying SDN in WSN, in which they define a controller called Flow Sensor based on the OpenFlow protocol. They have divided the network architecture into a two-tier hierarchy, as seen in [40]. The first part is the Top Level Overlay and has a logical sink with several virtual flow sensors. The second part is a bottom level cluster concentrating all the sensor nodes associated with each cluster. The Logical clustering performance has been examined in terms of delay, jitter, and packet loss ratio in different scenarios such as variant packet flow rate, different number of nodes per group, and different group size. Some advantages and disadvantages are:

- Advantage: This proposal uses logical clustering for sensor network applications. Furthermore, each sink of the logical link can do processing locally.
- Disadvantage: This investigation does not have a routing protocol for the logical-clustering.

De Gante et al. [41] propose an architecture with sensor nodes and a base station with a generic controller function. In other words, it is a typical WSN, including a controller (base station). Sensor nodes cannot make routing decisions, and the controller calculates the decision for the routes. The authors consider the QoS to be in the lower energy overhead incurred by the nodes. Some advantages and disadvantages are:

- Advantage: Unlike previous research, this work proposes the use of flow tables based on OpenFlow protocol.
- Disadvantage: It is a theoretical proposal; it does not have any simulation or testbed to prove the reduction of energy consumption.

Galluccio et al. define SDN-WISE [42], a software defined networking solution for wireless sensor networks. SDN-WISE attempts to reduce the amount of information exchanged between sensors and SDN controllers. The operation of the SDN-WISE sensor nodes is based on three data structures: the WISE state matrix, the accepted formula matrix, and the WISE flow table. In terms of QoS, Paolo Di Dio et al. [43] have introduced a mechanism to support differentiated levels of QoS in WSN through SDN-WISE. This mechanism includes procedures aimed at mitigating the congestion in the network through load balancing. Some advantages and disadvantages are:

- Advantage: Sensor nodes generate different types of flows (traffic), classified into k different priority classes, where C1 is the highest priority class, and Ck is the lowest priority class.
- Disadvantage: The flow table contains much information for nodes with limited resources. Therefore, these nodes (sensors and others) spend more energy to process and at the same time use more memory to save this information.

In [44], the authors propose a mobile agent-based architecture of MANET nodes for data transfer from the SDN environment. Regional data transmission through the mobile agent can reduce the registrations and modification of the flow table agent by routing autonomously. On the other hand, there is no quality control of the service. Some advantages and disadvantages are:

- Advantage: Using a mobile agent forwards only the externally-relevant information transmitted from the main node to the SDN controller. The local routing method is based on the AODV routing protocols, and out-of-region transmissions are tagged with an area id to identify which region they came from.
- Disadvantage: It is not clear when it uses mobile agents in the movement of nodes and can only transmit to the outside of each region when the mobile agent is in range of the main node, and it does not control the quality of service.

The authors in [45] propose an architecture to allow many controllers for a Software Defined Wireless Sensor Network. This architecture presents two types of identified nodes: an SDN controller and an enabled sensor device. It also defines the types of devices, the control flow, and the appropriate activities.

The SDN controller in this proposal uses and executes SPIN (Sensor Protocols for Information
latex. Some advantages and disadvantages are:

- Advantage: Routing protocols are required to maintain the network. In this proposal, security is also achieved in host-level nodes by the controller, and the nodes will not receive or forward any messages without the knowledge of the controller.
- Disadvantage: This model does not work for heterogeneous networks.

In [48] Qin et al. propose MINA (Multinetowrk Information Architecture), in which there is a reflective middleware layer with an IoT SDN controller. This IoT SDN controller incorporates and supports commands to differentiate the flow planning depending on the task level, multi-hop, and ad-hoc heterogeneous ways. It also uses and exploits network computation and Genetic algorithms as the routing protocol with multiple QoS constraints, which plans the flow (with multiple QoS constraints) to optimize using the opportunities available at present in the IoT network. Some advantages and disadvantages are:

- Advantage: This protocol uses a Genetic algorithm with multiple QoS constraints in heterogeneous networks to calculate the best route. The MINA SDN was implemented in a testbed, large-scale integration of electric vehicles, electric recharge sites, smart grid infrastructure, and a wide range of pilot users, such as targeted white Artemis Internet Energy and Arrowhead projects.
- Disadvantage: It does not specify the energy savings in the nodes and the entire network, and it is unlikely to use devices with limited features.

SDN-IoT concept and standardization efforts are still under development. However, some proposals related to QoS routing in IoT based on SDN have been addressed. In addition to the analyzed proposals, there are other mechanisms like Multi-task SDN [49], Mesh Flow [50], wmSDN [51], and Multi-controller mesh [52] which also use an OpenFlow based controller. However, these mechanisms do not contemplate an efficient energy-saving technique for the nodes, and control packet loss or latency. MINA SDN is the only protocol that presents a proposal for mesh networks (multinetowrk). Additionally, it uses genetic algorithms to meet multiple QoS constraints. Whereas the Rahmani et al. [40] proposal offers an interesting architecture because it permits the separation of the network by cluster, this allows for a more efficient SDN implementation due to the organization of the IoT network.

Other SDN architectures can be found in state of the art. In [53], authors introduce a new architecture for low-power wireless networks in SDN. This synchronous flooding (SF) architecture dynamically sets up SF protocols to provide reliable and low-latency performance when complex SDN control requirements are needed. In [54], a novel SDN architecture named VERO-SDN proposes a programmable routing control, a modular SDN controller, and an OpenFlow-like protocol. The approach of this architecture is to get reduced control overhead in long-range control channels and improve the communication quality in IoT scenarios with a wide range of coverage. Finally, in [55], a Multi-Protocol Software-Defined Networking Solution for the Internet of Things named MINOS is proposed, applying service awareness concepts and two configurable on-demand network protocols. This platform also offers a GUI with a personalized control panel and a real-time visualization tool.

IV. SBR: A NOVEL ARCHITECTURE OF SOFTWARE DEFINED NETWORK USING THE RPL PROTOCOL FOR INTERNET OF THINGS

This section describes an SDN architecture using the RPL protocol. The RPL protocol in-store mode allows the SDN controller to keep the flow tables updated using the control messages to manage, route the traffic and offer QoS. In the following subsections, the proposed SBR will be described, including its control and data planes and the construction of the topology needed to execute SBR.

A. SDN CLUSTER ARCHITECTURE FOR IoT

Some authors have proposed different architectures for IoT, such as [56], [57] and [7]. A key difference with the proposals described before is that SBR bases its design on SDN using clusters or domains [40], [58]. The proposed architecture was created to address several management areas, such as mobility and location. In addition, it provides reconfiguration capabilities according to the SDN paradigm, improving aspects focused on energy management and topology discovery, among others. Figure 3 shows the components involved in the proposed architecture.

B. CLUSTERS OR SDN DOMAIN

Each cluster or SDN domain comprises two types of nodes: a general node and a controller node (also named edge node). The general node can be a device with minimum characteristics, such as a sensor or any intelligent object. The edge node is the controller, and its hardware characteristics are more robust.

C. DOMAINS COMMUNICATION

The network is formed with various domains. SDN domains contain general nodes, connecting a single edge or gateway...
through UDP in a specific port, unique to each domain. Each domain has its controller or edge node (Figure 3). These gateways are connected via UDP using a different port to a single node with more processing capacity and acting as supervisor. This supervisor works as the root of the RPL network.

D. CONTROL PLANE

In the control plane, control messages are executed and sent to SBR by the RPL protocol, in which the objective function SIGMA [59] is used for selecting the best neighbor node. Sending control messages allows the controller to manage and ensure that the topology is stable. The control plane addresses the general nodes using ipv6 addresses. For inter-domain communication (developed in the edge nodes), IPv6 addresses and logical ports are used (Figure 3).

E. CREATION OF THE TOPOLOGY

Each cluster organizes the nodes in a DODAG (Figure 3). According to the rank value found in each of the parents, nodes of the DODAG take their closest neighbors as possible parents. Thus, the node with the lowest value is selected. However, each potential parent is a route to reach the root of the cluster. In this way, the root node of the DODAG knows the complete network to manage the routes in each available flow table. Thus, a network can encompass several DODAGs, which are identified by the following parameters:

| Parameters       | Description                                                                 |
|------------------|----------------------------------------------------------------------------|
| RPLInstanceID    | It is used to identify an independent DODAG set                            |
| DODAGID          | Identifier of a DODAG root                                                 |
| DODAGVersionNumber | This parameter increases with some specific events, such as the reconstruction of a DODAG. |
| Range            | Parameter to define node’s position                                         |

In the root, a structure stores the paths obtained in DAO messages called dao_routes. Routes export the operations add member route, and obtain a route to add a new hop to the path towards a specific node according to the information received in the DAO messages. In turn, to obtain a complete route to a node, respectively.

According to Sanmartin et al. [59], in an RPL protocol with the objective function SIGMA for an IoT environment, the root of the DODAG initially sends a DIO message. Any in-range nodes will receive this message, which will then propagate it downwards through the DODAG. Every time a node joins the DODAG (and spreads the DIO), it will send a DAO message addressed to the root to inform of its presence and to determine the available routes to reach it (See Figure 4). This process will be executed by using the SIGMA-ETX metric, which is based on the ETX metric. SIGMA-ETX metric is calculated using new optional fields added to the DAO, and these fields contain the ETX value in the sender and its local links to the neighbors.

F. DATA PLANE

In the data plane, the packets are handled according to the flows. A flow is a set of programmable packages (user-customizable), sharing specific properties indicated by a match in a flow table entry or a flow entry. The flow tables are updated by the controller through control messages every certain period according to the flow. In Figure 6, for example, the first row of the flow table defines a condition that will use node_W as the next-hop of all packets from node_X and have node_Y as their destination.

The second row defines a condition that will discard and stop further processing on all packets with a 32bit unsigned integer at the start of the packet body, whose value is lower than the 32bit unsigned match value X. The third row defines a condition that will discard all packets addressed to node_Z. There is also a rules field, each flow table entry can have...
multiple rules (see Figure 7), and there are two types of rules, Routing rules and Content rules. The Routing rules can be used to match the packet header fields with specific values. A routing rule contains the following fields:

- **Source Port**: The Transport Layer port from which the packet was sent. It is a Source Port Mask that specifies which bits from the source port will be compared.
- **Destination Port**: The Transport Layer port to which the packet was sent. A Destination Port Mask, which specifies which bits from the destination port will be compared.
- **Source Address**: Determines the IPv6 address of the node that sent the packet. A Source Address Mask, which specifies which bits from the source address will be compared.
- **Destination Address**: The IPv6 address of the node from the packet, was sent. A Destination Address Mask that specifies which bits from the destination address will be analyzed.

The Content rules can be used to match the content of the packets, allowing comparisons of arbitrary data types in the body of the packet. A Content rule has the following fields:

- **Operation**: Specifies the operation will be performed between the values obtained from the packet and the value stored in the rule. It can be Less Than (<), Greater Than (>), or Equal To (=).
- **Offset from start of content**: Determines where the data is located in the packet’s body (measured in bytes).
- **Data size**: Determines how many bytes to read from the packet’s body and the number of bytes of the Match Value field.
- **Signedness**: Determines whether the values will be interpreted as signed or unsigned values.
- **Match value**: The amount to compare. The number of bytes in this field depends on the value of the Data size field. If no match is found for a packet in the flow table of a node that receives it, the packet will continue its standard processing by the IPv6 stack as if there was no flow table. For RPL SIGMA, this means that it will be sent to either the destination node if it is a local link to the current node or the preferred parent in the other case.

### V. NETWORK SIMULATION DESIGN

The simulation is based on the IPv6 architecture and uses IEEE 802.15.4 at the PHY and MAC layers to form LLNs. On top of that, it uses 6LoW-PAN, RPL, and IPv6 to provide end-to-end two-way communication to each node (See Table 3). Simulations are implemented in the Cooja simulator to test scenarios with multiple nodes per cluster, as shown in figures 8 and 9 (Scenarios A with clusters of 10 nodes and B with clusters of 5 nodes are displayed).

### TABLE 3. Summary of simulation parameters.

| Parameters                      | Value                                                                 |
|---------------------------------|----------------------------------------------------------------------|
| Area                            | 230m x 230m                                                          |
| Boot delay per node             | Random between 0 and 20 seconds                                      |
| Sensing interval                | Random between 0 and 50 seconds                                      |
| Number of nodes                 | 3 clusters with 3, 5, 10 nodes each                                  |
| TX Range                        | 50m                                                                  |
| Network Protocol                | Modified RPL (SBR)                                                  |
| Standardized Objective Functions| SIGMA                                                                |
| Standard                        | 802.15.4                                                            |
| Topology                        | 1 root controller with 3 cluster gateways and N nodes in each cluster|
| Number of packets               | 200                                                                 |
| Model of radius                 | UDGM                                                                |

The topology of the simulated network consists of a root node that acts as a central controller and as the root of the RPL...
Instance, with three more secondary controllers (we will call these gateways) assigned to several sensing nodes. Each sensing node connects to its assigned gateway and periodically sends data based on the sensing interval, a random number between 0 and 50 seconds for each node. Each sensing node is assigned to a gateway at the configuration time, periodically sends simulated “sensed” data to its gateway. The gateway nodes periodically send packets to the root controller to simulate control messages being exchanged between them.

Some variations of the rules were implemented, as seen in Figure 11. In this scenario, all sensing nodes send a packet every 5 seconds, but node 9 drops every second packet it receives to simulate deterministic packet loss. Default RPL sets node 9 as the parent node for nodes 3, 6, and 11, thus causing a bottleneck. Through SBR, a rule is added to the flow table of nodes 3, 6, and 11 to redirect all traffic through node 7 instead of node 9, avoiding the traffic bottleneck.

In Figure 12, node 14 is overloaded by being the preferred parent of several leaf nodes. Using SBR rules in nodes 22, 23, 10, and 27, it is possible to balance the traffic and spread it through a broader area in the network. This process avoids the problem of overloading a node’s queue. The rules used were all forward rules from nodes 22 to 25, from 23 to 20, from 10 to 15, and 27 to 13.

VI. RESULTS
Multiple simulations were implemented in the Cooja simulator to test scenarios with various nodes per cluster (3, 5, and 10). Each node in the simulation runs the Contiki operating system and uses the SBR over 6LoWPAN over the 802.15.4 protocol.

In the scenarios presented in section 5, the simulation was performed with two approaches, SDN using RPL (SDN-RPL) and SDN using RPL-SIGMA (SBR), where each scenario had 3 clusters. Different rules (Drop, Forward) were tested in the nodes through messages sent from each edge, which allowed redirecting the traffic depending on the state of the network, the congestion, and the administrator/controller.

A. LATENCY OF THE NETWORK
The result of latency, packet loss, and energy consumption is presented. Figure 13-A of 200 packets sent through the architecture, SBR had a mean delay of 1.75 seconds, 11% less than SDN-RPL. It can be seen, on average, SBR had a much lower
delay than the total latency. This situation is because when the simulation was started, as explained in Figures 15 and 16, there was a traffic bottleneck. This problem disappeared with the controller’s changes that consisted of redirecting traffic to another node in the network through messages that directly affected the flow table of the nodes involved.

Fig.13-B shows the total energy consumption using both approaches. SBR had a 20% lower consumption during the simulation, slightly less than SDR. This situation indicates that the control messages allow for a smaller energy footprint due to nodes having the ability to avoid additional processing in scenarios where bottlenecks are frequent, in contrast to nodes that cannot rapidly perform load balancing, as in the examples in Figure 11 and Figure 12. On the other hand, Figure 14 shows that the confidence intervals of 95% for packet delays do not overlap each other, as seen by the red line, which hints at a solid statistical significance.

B. PACKET DELIVERY

Packet loss in the network also displayed a better performance with SBR (52%) after applying drop and forward rules from scenario D in figure 12, whereas SDN-RPL had a packet loss of 70%. Nodes that sent packets at the start of the simulation always experience a higher amount of packet loss due to bottlenecks. Then the controller starts sending control messages to balance the network.

Figure 16 shows that the confidence intervals of 95% for packet loss do not overlap each other, as seen by the red line, which hints at a solid statistical significance. Packet delay also shows a similar statistical significance hint.

VII. CONCLUSION

In this paper, a novel architecture of SDN was proposed using the RPL protocol in the control and data planes for IoT architectures. This proposal solves many quality of service problems, reduces energy and memory consumption, packet loss, and extends the lifetime of the IoT network having WSN. The proposed SDN architecture includes the RPL protocol with the SIGMA-ETX metric (RPL-SIGMA).

An architecture working with clusters or IoT domains is also included, then SDN can be used to efficiently manage the IoT network. The SDN architecture using the RPL protocol with the objective function SIGMA for IoT (SBR) exploits the effectiveness of RPL to create and update network topologies. Additionally, RPL messages took advantage of this functionality so that the control plane works efficiently.

Finally, a comparison between SBR and SDN-RPL was performed. In terms of energy consumption, SBR had a 20% lower consumption than SDR. Packet loss in the network also showed a better performance with SBR (52%), whereas SDN-RPL had a packet loss of 70%. The simulations indicated in general that SBR has a better performance concerning packet loss, latency, and energy consumption over an architecture based solely on IoT domains.

REFERENCES

[1] ITU Internet Reports 2005: The Internet of Things, ITU Internet Reports, Geneva, Switzerland, 2005.
[2] Z. Yan, P. Zhang, and A. V. Vasilakos, “A survey on trust management for Internet of Things,” J. Netw. Comput. Appl., vol. 42, pp. 120–134, Jun. 2014.
[3] S. Y. Hashemi and F. S. Aliee, “Dynamic and comprehensive trust model for IoT and its integration into RPL.” J. Supercomput., vol. 75, no. 7, pp. 3555–3584, Jul. 2019.
[4] M. Huang, A. Liu, N. Xiong, T. Wang, and A. V. Vasilakos, “An effective service-oriented networking management architecture for 5G-enabled Internet of Things,” Comput. Netw., vol. 173, May 2020, Art. no. 107208.
[5] G. Sun, Z. Xu, H. Yu, X. Chen, V. Chang, and A. V. Vasilakos, “Low-latency and resource-efficient service function chaining orchestration in network function virtualization,” IEEE Internet Things J., vol. 7, no. 7, pp. 5760–5772, Jul. 2020.
[6] M. Wazid, A. K. Das, V. Bhat K, and A. V. Vasilakos, “LAM-CIoT: Lightweight authentication mechanism in cloud-based IoT environment,” J. Netw. Comput. Appl., vol. 150, Jan. 2020, Art. no. 102496.
[7] N. A. Jagadeesan and B. Krishnamachari, “Software-defined networking paradigms in wireless networks: A survey,” ACM Comput. Surv., vol. 47, no. 2, pp. 1–27, 2014.
[8] M. Ndiaye, G. P. Hancke, and A. M. Abu-Mahfouz, “Software defined networking for improved wireless sensor network management: A survey,” Sensors, vol. 17, no. 5, p. 1031, 2017.
[9] H. I. Kobo, A. M. Abu-Mahfouz, and G. P. Hancke, “A survey on software-defined wireless sensor networks: Challenges and design requirements,” IEEE Access, vol. 5, pp. 1872–1899, 2017.
[10] E. Ahmed, I. Yaqoob, I. A. T. Hashem, I. Khan, A. I. A. Ahmed, M. Imran, and A. V. Vasilakos, “The role of big data analytics in Internet of Things,” Comput. Netw., vol. 129, pp. 459–471, Dec. 2017.
[11] O. Gnawali and P. Levis, The ETX Objective Function for RPL, IETF Internet Draft: Draft-Gnawali-Roll-Etxof-01. Accessed: Feb. 15, 2018. [Online]. Available: http://tools.ietf.org/html/draft-gnawali-roll-etxof-00
PAUL SANMARTIN (Member, IEEE) received the M.Sc. degree in computer science and the Ph.D. degree in computer science from the Universidad Del Norte, Colombia. He is currently an Associate Professor with the Faculty of Engineering, Universidad Simón Bolívar. His research interests include quality of service on the Internet, the IoT, WSN, routing protocols, NGN, and telematics applications.

KAREN AVILA is System Engineering and Ph.D. student in Computer Science at Universidad del Norte in Barranquilla, Colombia. She is currently a Research Assistant and a Professor with the Department of Systems Engineering, Universidad del Norte. Her research interests include the IoT, WSN, routing protocols, and software development.

SEBASTIAN VALLE is currently a Systems Engineer and a Magister (C) with the Universidad del Norte, Barranquilla, Colombia. His research interest includes wireless sensor networks.

JAVIER GOMEZ received the M.S. and Ph.D. degrees in electrical engineering, Columbia University. He is currently the Co-Director of Wireless Research Laboratory, UNAM. His research interests include performance modeling, analysis and design of algorithms, and protocols for wireless dd hoc and sensor and mesh networks.

DALADIER JABBA (Senior Member, IEEE) received the Degree in system engineering, the M.Sc. degree in computer engineering, and the Ph.D. degree in computer science and engineering from the University of South Florida, Tampa, FL, USA, and the M.Sc. degree in computer science from UNAB, Colombia. He is currently an Associate Professor with the Department of Systems Engineering, Universidad del Norte, where he is also a member of the Research Group on Computer Networks and Software Engineering (GReCIS). Before joining the Universidad del Norte, he belonged to the Research, Development and Innovation Department, as an International Resources Manager (Research and Development). He has published more than 40 articles in different journals and ten publications between books and book chapters. He works in the projects of university–industry interaction. He was with the National Committee that designs templates in order to develop the questions in the tests named Saber Pro for students of final semesters in Colombian Universities. He was part of the group of researchers belonging to the Technical Secretariat of the Universidad del Norte, supported by the “Mission of Wise Men” appointed by the Colombian Government, in 2019, in the focus of Industry 4.0.

***