Microprocessors and Microsystems 69 (2019) 79–91

Contents lists available at ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro

Effects of RPL objective functions on the primitive characteristics of mobile and static IoT infrastructures

Bardia Safaei[a], Ali Asghar Mohammad Salehi[a], Amir Mahdi Hosseini Monazzah[b], Alireza Ejlali[a,*]

[a] Department of Computer Engineering, Sharif University of Technology, Tehran, 11155-11365, Iran
[b] School of Computer Science, Institute for Research in Fundamental Sciences, Tehran, 19395-5531, Iran

ARTICLE INFO

Article history:
Received 15 October 2018
Revised 20 May 2019
Accepted 26 May 2019
Available online 4 June 2019

Keywords:
Internet of Things
Mobility
IPv6 Routing Protocol for Low-power and lossy networks
Objective function

ABSTRACT

The emergence of mobile IoT applications in recent years and the challenge of routing in their infrastructures have motivated scholars to propose appropriate routing mechanisms for such systems. Meanwhile, the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL) is the standard routing protocol for IoT infrastructures. Nevertheless, RPL was mainly designed to comply with the primitive requirements of static IoT applications and it behaves poorly in confronting with the severe alterations in mobile conditions. The most important factor for such a poor behavior in mobile applications is the inappropriate design of OFs, which determine RPL's routing policies in the network. Therefore, the functionality and efficiency of the existing OFs should be carefully studied in order to determine their downsides and to resolve them in the future OF designs. Accordingly, in this paper, due to lack of information on how OFs affect the primary requirements of IoT infrastructures in mobile and static infrastructures, we have compared and evaluated the effect of state of the art OFs on the functionality of the nodes and the network from different perspectives. Our evaluations on fixed-node IoT infrastructures have shown that OFs could affect the power consumption by up to 80%, while their impact on reliability and E2E delay would be as high as 54% and 75%, respectively. Further observations have proven that mobility would reduce OFs contribution in power consumption by more than 10% in comparison with static IoT infrastructures, while it can increase their influence on reliability and E2E delay by nearly 15% and 16%, respectively.

1. Introduction

Internet of Things has the potential to ignite another industrial revolution because of its ability to surround every aspect of human's daily activities. IoT is an emerging topic of technical, social, and economic significance, which different involved associations have not yet converged to an agreement on its standard definition. According to Safaei et al. [1], IoT is a system comprising a communicative infrastructure which establishes a connection between an enormous amount of smart, identified, and low-power embedded devices using Internet-based communication technologies without human intervention.

It has been anticipated that there will be more than 9 devices per person at the end of 2025 [1]. Aggregation of such an enormous amount of connected devices, demonstrates that IoT has the potential to effect human lives and the global economy from every aspect. McKinsey Global Institute® has published a report, suggesting that the financial impact of the IoT on the global economy may reach as much as \$3.9 to \$11.1 trillion dollars at the end of 2025 [2].

The main reason for such an impact is the broad range of supported applications by IoT. The IoT applications could be categorized into two domains: 1) Static, and 2) Mobile. Static IoT applications refer to a group of applications containing a number of fixed-nodes in the environment; Applications such as smart buildings, industrial automation, environment monitoring, and emergency response to natural and man-made disasters are placed within this category. Unlike fixed-node IoT infrastructures, in mobile IoT applications, due to movements of the embedded devices, the network topology is time-variable and thus, the link breaks and alterations in the network are unavoidable. The location sensing and sharing of location information in smart vehicles, and commodity tracking are a number of well-known mobile IoT applications.

IoT infrastructure mainly relies on communication of many resource constraint embedded devices with low processing power,

* Corresponding author.
E-mail addresses: bsafaei@ce.sharif.edu (B. Safaei), mohammadsalehi@ce.sharif.edu (A.A. Mohammad Salehi), monazzah@ipm.ir (A.M. Hosseini Monazzah), ejlali@sharif.edu (A. Ejlali).

https://doi.org/10.1016/j.micpro.2019.05.010
0141-9331/© 2019 Published by Elsevier B.V.

memory, energy (battery power), data rate, and transmission range, in wireless mediums. Although IoT and its predecessor Wireless Sensor Networks (WSN) have fundamental distinctions, e.g., the IP based Internet connections and the essence of identifying the sources of transmissions in IoT, the high packet loss nature of wireless environments has placed both of these technologies as a subset of larger wireless environments known as Low Power and Lossy Networks (LLN) [3–6]. Communication of such an enormous amount of connected devices in LLNs, arises many considerations. Accordingly, different levels of IoT abstraction are facing major challenges, which are substantial to be managed. Among different layers of an IoT infrastructure, the network layer and relatively its routing protocols play an important role in providing a low cost and reliable communication in order to deliver data packets from source nodes to their destination(s).

In this regard, in 2008, the Internet Engineering Task Force (IETF), established the Routing over Low-power and Lossy networks (RoLL) working group to determine the prerequisites, design and implement a routing mechanism, which is applicable for constraint embedded nodes and satisfies the requirements of IoT systems. During 2009 to 2010, the group published a few documents, which included the routing requirements for four specific IoT applications, i.e., Urban [7], Industrial [8], Home automation [9], and Building automation [10]. The outcome was **R**outing **P**rotocol for **L**ow-power and lossy networks **(RPL)**, which was standardized and delivered to the public on march 2012 as an RFC6550 [11].

RPL is a gradient, proactive, IPv6 based routing protocol aiming to build a robust multi-hop mesh topology over lossy links with minimal state requirements [11]. It could deliver packets in different traffic patterns, i.e., point-to-point (P2P), point-to-multipoint (P2MP) or multipoint-to-point (MP2P), which is an important issue in the concept of routing in IoT systems. This distance-vector protocol is highly flexible and it could be adopted by various applications to fulfill their different network requirements. RPL is widely used in real world IoT infrastructures. The Cisco Resilient Mesh (formerly known as CG-Mesh) is a well-known IoT network stack which employs RPL for establishing the routing in smart grid applications [12].

The fundamental component, which provides the flexibility in RPL is the Objective Function (OF). OF is responsible for defining how embedded nodes select their path in order to forward data packets in an optimized manner according to the application's requirements. An OF determines how RPL nodes interpret one or more node/link metrics (e.g., node's residual energy, throughput, latency, link reliability), into a Rank; A value, which approximates the node's distance from the coordinator node (also known as a root or sink). A metric is a quantity, which is exploited to calculate the path cost and consequently the value of the rank. Typically, link and node metrics are additive [13].

OF0 [14] and Minimum Rank with Hysteresis Objective Function (MRHOF) [15] are a couple of the most well-known OFs, which are used in RPL based IoT infrastructures. OF0 is the basic OF in RPL, which does not exploit any node/link metrics and simply determines the preferred path based on the minimum hop count. On the other hand, MRHOF is more complicated and it is able to consider different node/link metrics to compute the nodes rank. These OFs could not fulfill every aspect of different IoT applications. Hence, in recent years there have been a number of studies on providing novel OFs to meet specific IoT characteristics, e.g., reliable communication, low power consumption, and longer system life time [16,17]). RPL was intended to be employed in static IoT applications and its standard deployment (which is introduced in [11]) in mobile networks performs poorly. The lack of compatibility and slow response to the topology alterations in the standard RPL makes it a highly inefficient and potentially an impractical routing protocol for mobile IoT infrastructures [18]. Hence, several studies have provided modified versions of RPL to support mobility [18–21]; Nevertheless, to the best of our knowledge, implementing an appropriate OF, which should be specifically designed for mobile RPL based IoT infrastructures is still an open challenge with no previous efforts. Prior to designing an OF for mobile IoT applications, it is essential to highlight and analyze the weakness of state of the art OFs from different perspectives and compare their functionality in static and mobile conditions.

Hence, in this paper, due to lack of information on how OFs affect the primary properties of IoT infrastructures in fixed-node and mobile scenarios, we have compared three well-known OFs, i.e., OF0, MRHOF, and Fuzzy Logic Objective Function (OFFL) [22] from different perspectives. This includes power consumption, End-to-End (E2E) delay, reliability, and topology control overhead. The accomplished study in this paper would be beneficial for the readers from two points of view: 1) It would pave the way for scholars to realize which of the mentioned areas are more affected by OFs, and 2) It conducts a set of experiments to explore the behavior of state of the art OFs in static and mobile IoT infrastructures.

In order to conduct a set of comprehensive evaluations, we have exploited Cooja simulator [23]. In this regard, we have considered different scenarios in mobile and static IoT infrastructures to illustrate the contribution of OFs in the behaviour of different IoT characteristics. Our simulation results have shown that in case of a static network with fixed nodes, OFs could affect the power consumption by up to 80%, while their impact on reliability, E2E Delay, and control overhead would be as high as 54%, 75%, and 50% respectively. On the other hand, further observations have proven that a mobile IoT infrastructure would reduce OFs contribution in the power consumption by more than 10% compared with static IoT infrastructures, while they can increase OFs influence on reliability, and E2E delay by nearly 15% and 16% respectively. In addition, mobility has increased the impact of OFs on the control packet overhead by more than 8%.

The rest of this paper is organized as follows: Section 2 represents a background on RPL and its structure. In Section 3, the previously proposed OFs have been surveyed and the built-in objective functions of RPL along with the novel OFFL is described in detail. Comparison strategies have been discussed in Section 4. System setup and experimental results are stated in Section 5. Finally, the paper is concluded in Section 6. A summary of notations that are exploited in the paper has been presented in Table 1.

Table 1
Summary of notations in the article.

Notation	Description
RPL	IPv6 Routing Protocol for Low-power and lossy networks
OF	Objective Funcion
OF0	Objective Function Zero
MRHOF	Minimum Rank with Hysteresis Objective Function
OFFL	Fuzzy Logic Objective Function
E2E	End-to-End Delay
DAG	Directed Acyclic Graph
DODAG	Destination Oriented Directed Acyclic Graph
LBR	LLN Border Router
DIO	DODAG Information Object
DIS	DODAG Information Solicitation
DAO	Destination Advertisement Object
MoP	Mode of Operation
OCP	Objective Code Point
ETX	Expected Transmission Count
PDR	Packet Delivery Ratio
TI	Transmission Interval
LPM	Low Power Mode
LSR	Link Success Rate

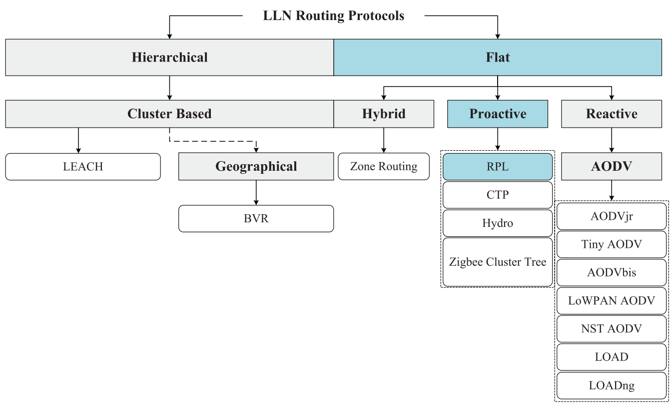

Fig. 1. Low power and lossy networks routing protocols classification.

2. Background on RPL

Considering the rapid growth in number of connected devices, the variety of LLN and IoT applications, and the pervasive nature of these two environments, different international standardization bodies and academic institutions, have developed and standardized novel routing protocols to meet the requirements of different IoT applications. Basically, as it has been depicted in Fig. 1, existing routing protocols for low power and lossy networks could be broadly classified into two major groups: 1) Flat, and 2) Hierarchical [24]. Flat protocols are composed of many nodes, which have the same contribution in the network. An specific advantage of such protocols is their minimal imposed overhead for maintaining the infrastructure among the communicating nodes [25]. The other group of protocols are hierarchical protocols, which define an structure for the network. In these protocols nodes are organized in clusters to gain scalability and efficiency.

The Flat category itself could be classified into three subcategories: 1) Proactive, 2) Reactive, and 3) Hybrid routings. Proactive protocols are a group of routing mechanisms in which, each node builds its own routing table to be able to send packets to the destination. Any alterations in the network will lead into transmission of update packets in order to maintain the network's stability. Thus, when a node decides to send a packet to a specific destination, there wouldn't be any extra delay on route discovery procedure. However, maintaining the routing information up-dated, imposes high resource consumption for the IoT nodes in the LLN (e.g., power, memory, and bandwidth) [25]. Collection Tree Protocol (CTP) [26], Hydro [27], and Zigbee cluster-tree [28] are a few samples of proactive protocols.

Unlike the proactive protocols, in reactive protocols, nodes begin the route discovery procedure only when there is a demand for it; This is why this group of protocols are also known as on-demand routing protocols. This feature eliminates the need for storing the updates on the nodes and consequently reduces the memory usage, but this process imposes delays due to route discovery operation, which is caused by the absence of requested routes. Ad-hoc On-Demand Distance Vector (AODV) routing protocols are a wide family of reactive routing protocols in which routing information is exchanged between distributed nodes in the system [29]. Another group of flat routing protocols, consider the positive aspects of the two previous groups to form a new class which is called hybrid routing protocols.

On the other side, there exist hierarchical routing protocols, which are also known as the clustering based protocols. These protocols can extend the lifetime of the network by reducing the consumed energy of the routing mechanisms. Geographic routing protocols, which are designed to enable nodes to accomplish the route discovery procedure based on the knowledge of location, are a subset of clustering based protocols and Beacon Vector Routing (BVR) is a good example for this group of routing protocols [30].

The above mentioned routing protocols are designed and optimized for collecting information or they specifically focus on point-to-point traffic patterns in homogeneous networks. It is worthy to know that many of these protocols do not fulfill the IoT requirements (e.g., power consumption). In addition, the pervasive nature of IoT and LLNs requires a scalable, comprehensive, flexible, multi-hop and low power routing protocol which supports different traffic patterns in the network and meets the pre-requisites of the IoT nodes. Further more, employing a multi-hop routing mechanism in IoT infrastructures is essential and it cannot be replaced with the emerging low-power, low bit rate and long-range radio technologies, e.g., LoRa/LoRaWAN [31] and Sigfox [32]. Because, these technologies are typically employed in applications, where the data transmission is done towards a base-station (in form of a star topology), while in many pervasive IoT applications, due to the harsh, dynamic and lossy nature of the wireless mediums and the mesh-based structure of the communications, the multi-hop routing may be the only option [33]. Accordingly, RPL seems to be the right choice for IoT infrastructures.

Basically, RPL is a distance-vector routing protocol, which is capable of operating on top of several link layer technologies, e.g., IEEE 802.15.4 PHY and MAC [11]. It was designed to be highly adaptive to network alterations in case of inaccessibility at default routes, specifically in collection based applications were hundreds of thousands of fixed smart devices, measure environmental parameters from the physical world and send them to a collecting point [22]. This source routing protocol constructs a tree shape topology known as Directed Acyclic Graph (DAG) rooted at one or more sink nodes (also known as LLN Border Router or LBR in case of connecting the DAG to the Internet). In this structure, paths connect end-nodes (leaves) to the sink nodes. A DAG which is composed of only a single root is called a Destination Oriented Directed Acyclic Graph (DODAG).

The construction of DODAG is being affected by the defined routing metrics in the RPL's OF. The OF is responsible for optimizing a single or a set of path costs according to the intended application requirements. An application could be composed of several RPL instances with different and potentially inconsistent constraints with each containing one or more DODAGs. RPL divides the DAG into one or more DODAGs and dedicates a $DODAG_{ID}$ to each of them to distinguish them in an RPL instance (which has a specific $RPL_{Instance\_ID}$ itself). The DODAG is also assigned with a $DODAG_{Version\_Number}$ which is increased by the root every time the DODAG is rebuilt. The unique version of the DODAG is identified via the triple values of $RPL_{Instance\_ID}$, $DODAG_{ID}$, and $DODAG_{Version\_Number}$. The engaged nodes in RPL, exchange four types of ICMPv6 control messages to construct the DODAG and consequently to route data packets from source nodes to the destination nodes. The structure of RPL and its creation has been illustrated in Fig. 2.

The construction of the DODAG is initiated via DODAG Information Object (DIO), where the root broadcasts this control message to its neighbours to inform the minimum rank value. DIO provides information about how nodes can discover an RPL instance, learn the configuration parameters of the intended instance, determine a set of nodes as their parent set and finally how to maintain the DODAG. Upon reception of the DIO in a node, according to the provided information in the DIO message (specially the DAG metric container and DODAG configuration option) and the pre-defined OF, it chooses a number of neighbour nodes as its candidate parent set and further selects one of them as its preferred parent. Furthermore, the node computes its rank by increasing its parents rank monotonically. After completion of the rank calculation, the node

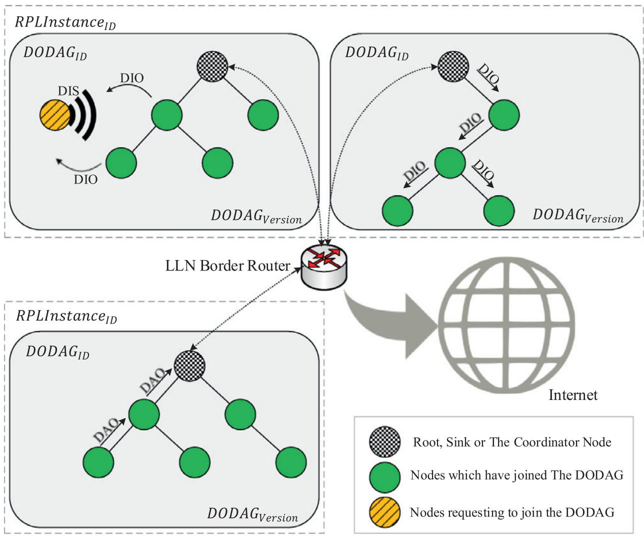

Fig. 2. RPL structure.

propagates its rank with all the updated information to its neighbours and this procedure iterates until the DODAG creation. DIO messages are broadcasted with an exponentially increasing interval defined by the trickle algorithm to mitigate the control overhead as well as the power consumption. A node can join a DODAG by transmitting a Destination Information Solicitation (DIS). Any of the neighbour nodes which receive a DIS message, will send a DIO message unicastly to the solicitating node. Consequently, the requesting node would choose its preferred parent according to the received DIO messages from the neighbour nodes.

The aforementioned control messages are the core components for the upward routing (a route from non-root nodes to the root) and relatively the MP2P traffic flows. However, RPL supports P2MP and P2P traffic patterns which necessitates the downward routing (a route from root to the non-root nodes) capabilities. Destination Advertisement Object (DAO) is responsible for enabling downward routes in RPL. There are two Modes of Operation (MoP) in RPL, i.e., storing, and non-storing, which determine weather the intermediate nodes are able to maintain the routing information carried by the DAO control message or not, respectively. It should be noted that, upon an explicit request from the sender node or occurrence of an error, an acknowledge packet will be sent from the receiver side through a Destination Advertisement Object Acknowledgement (DAO-ACK) message back to the transmitter.

3. Objective functions and related studies

The OF's duty is to provide an optimized path selection and topology construction in an RPL instance based on the specifications in a broad range of applications and network designs. An OF assists a node to join the appropriate DODAG with its convenient version. In addition, the OF determines how RPL nodes interpret one or more node/link metrics into a Rank [11]. A set of supported node/link constraints and metrics has been specified in [13]. To reach a valid RPL construction, OF specifies three relevant sets: 1) Neighbour Set, 2) Parent Set, and 3) Preferred Parent Set [34]. The DIO message carries the important information about OF in its configuration option field (which is part of the DIO options). In the configuration option section, the Objective Code Point (OCP) is responsible for identifying the OF. The RPL standard does not enforce any adoption of certain OFs or metrics for path selection and even separates the OF implementation from the fundamental specifications of RPL. Indeed, RPL provides the opportunity for engineers to design and implement a customized routing protocol in a flexible manner considering different characteristics of the intended application.

OF0 and MRHOF are the default objective functions implemented in the standard RPL routing protocol [14,15]. Nevertheless, there have been several studies on implementing new OFs with specific purposes (e.g., performance, power management, reliability, energy, load balancing, and etc). Accordingly, authors in [22,35,36] have tried to provide a better path selection in terms of energy efficiency by employing the node's remaining energy along with other metrics in the structure of their proposed OF. In a related study, to provide an energy efficient and congestion-aware OF for Advanced Metering Infrastructures (AMI), authors have proposed a novel routing metric which is called ECRM [37]. This routing metric considers both, the residual energy of the nodes and occupancy of their buffers to provide more reliable and power efficient communications in AMI applications. A number of energy-aware routing metrics which have been recently employed in RPL are surveyed in [38].

The energy depletion in an IoT node could lead into Black Hole Problem which is referred to those of situations, where a part of the network's tree, which is connected to that node (and it does not have any other available options to connect to), will loose its connection to the rest of the network. Therefore, in such systems, the first node which turns down, determines the lifetime of the entire network. Accordingly, many scholars have put their efforts on designing OFs with the aim of mitigating the nodes' energy consumption to prolong the network's lifetime [22,39,40]. SEEOF is one of the OFs which is proposed in [41]. SEEOF employs a novel routing metric that is called the Estimated Remaining Life Time (ERLT). ERLT not only uses the residual energy of the nodes, but is uses their energy drain rate to predict their probable death time and avoid selecting those of nodes with lower remaining up-time. Load balancing is another approach to gain more lifetime in the network. The Load-Balanced OF (LB-OF) was proposed in [42] to limit and balance the number of acceptable child nodes by each of the parents. The Expected Life Time (ELT) is another routing metric proposed in [43], and [44], which estimates the energy bottlenecks with high probability of running out of energy.

In another group of studies, the main design objective of the OFs is to reduce the control packet overheads in the network. The authors in [45] have came up with a novel OF, which introduces a stability metric that considers the rate of the transmitted control packets in each of the nodes in a passive manner. Providing a reliable communication is another important parameter in IoT infrastructures. Therefore, a number of studies try to enhance the reliability of the routing by using the Expected Transmission Count (ETX) metric [22,36,46–48], or by preventing congestion and packet drops in the buffers of the nodes through employing novel metrics such as queue utilization factor (QU-RPL) [49]. In addition to utilization factor, which determines the amount of occupancy in the buffer, QU-RPL exploits the hop distance to the LLN Border Router (LBR) to provide higher amounts of Packet Delivery Ratio (PDR) compared to the standard RPL. Similarly, authors in [37] has used the queue utilization index and the residual energy of the nodes to establish reliable and energy efficient communications in the network. In addition to the mentioned criteria, in numerous IoT applications such as Remote Health Monitoring Systems (RHMS), the latency is on a high consideration. The authors in [50] have proposed a novel OF which employs a routing metric called the Power-Delay Product (PDP) and tries to establish a trade-off between both, the power consumption and the E2E delay simultaneously to reduce the latency while improving the power consumption of the nodes.

Despite the existence of different kinds of proposed OFs for various IoT applications, the design, implementation, and employment of more comprehensive OFs and routing metrics, which can meet the emerging IoT applications' requirements (especially the emerging mobile IoT applications), is still an open research area. In addition to OF0 and MRHOF which are being frequently used in contemporary IoT applications, QoS-Aware Fuzzy Logic Objective Function (OFFL) is one of the most successful OFs among the previous studies [22]. OFFL is a novel OF, which exploits a set of node/link metrics and combines them to provide a configurable routing decision based on the fuzzy parameters and to improve the overall Quality of Service (QoS). In the following, the structure of these OFs has been described.

3.1. Objective function zero

The OF0 was designed with the aim of connecting an RPL node to the DODAG, which provides connectivity to a larger set of nodes and would pave the way for that node to forward its data packets to the root via the minimum hop count [14]. One of the challenges in OF0 is the issue of load balancing, which has not been considered in its core; Because each node routes its upward traffic through the preferred parent without any attempt to distribute them among the existing candidates in the preferred parent set. Nodes switch their parent only when there is a problem on the link connecting them to the parent node. The overall process of path selection begins with the calculation of a variable known as $Rank_{Increase}$ for every link. RPL requires four parameters to be able to calculate $Rank_{Increase}$ according to Eq. (1) [14].

$$R_{Increase} = ((R_f * R_p) + R_s) * R_m \tag{1}$$

Where R_f ($Rank_{Factor}$) is used to differentiate the type of the links in the same DAG (e.g., powered links over battery-operated or high-speed over low-speed links) by multiplying the influence of link specifications in computation of $Rank_{Increase}$ [14]. R_p ($Rank_{Step}$) is responsible for determining the amount of rank increment along a certain path in the DODAG. Another parameter in Eq. (1) is the R_s ($Rank_{Stretch}$), which is used to provide the ability for selecting at least one feasible successor and consequently to maintain the path diversity in the network. Finally, there is R_m ($Rank_{MinimumHopIncrease}$), which is a variable parameter and represents the units of which the $Rank_{Increase}$ is expressed with it. According to Eq. (2), to obtain the value of a rank for a node, the calculated $Rank_{Increase}$ should be added to the value of its preferred parent rank [14]. It is worthy to mention that in order to prevent loops in the RPL, nodes are only permitted to select those of nodes with lower ranks (compared to themselves) as their parents.

$$R_{Node} = R_{Parent} + R_{Increase} \tag{2}$$

3.2. Minimum rank with hysteresis objective function

The second OF which is implemented in the standard RPL and is more complicated than OF0, is MRHOF [15]. MRHOF has been developed to select a path by exploiting hysteresis. The term hysteresis refers to a relation in physical systems which their current condition depends on their historical state. The use of hysteresis enables RPL to prevent fluctuations in the parent selection and thus provides more stability in the network. This is due to the nature of parent selection in MRHOF which the current parent will be switched, if the path cost through the new parent is less than the current path cost, considering a predetermined threshold value. The parent selection has to be triggered every time the cost of an existing path in the node's neighbour table is changed or a new entry for a new node has been allocated in the neighbour table. If none of the above were met, the path selection would be repeated after a timeout determined by the trickle timer.

Hop count, ETX, and Latency are the metrics supported by MRHOF [15]. Latency as a metric could be exploited in applications with high demand for real-time responses, while ETX could be employed to improve reliability in terms of PDR. Only one node/link metric could be exploited by MRHOF and those metrics have to be additive. The path cost calculation will be initiated for each of the candidate nodes maintained in the neighbour table to detect the minimum value. According to Eq. (3), each node should compute its path cost ($Path_{Cost}$) to all of the existing nodes in its neighbour list by adding two parameters: 1) The node/link metric to that neighbour ($Link_{Cost}$), and 2) The metric value carried by the DIO metric container and sent by the neighbour ($Neighbour_{Cost}$) [15]. If any of the nodes were not able to accomplish this task, none of them should consider the other one as their preferred parent [15]. Moreover, if the calculated value was less than a predefined threshold, again the intended node should not select any node as its parent.

$$Path_{Cost} = Link_{Cost} + Neighbour_{Cost} \tag{3}$$

By default, the metric that has been defined in the structure of MRHOF is ETX. It refers to the minimum expected number of packet transmissions to a certain destination in order to reach a successful transmission. According to Eq. (4), ETX is equal to the inverse of PDR [19]. This relation indicates that a path, which benefits from high PDR, will provide lower ETX and thus, it could be selected as the preferred path with higher probability. The amount of PDR for a link ($Link_{PDR}$) could be computed by dividing two values respectively: 1) Total Number of Successful transmissions from one end to another end of the link ($Trans_{Success}$), and 2) Total number of transmissions from one end to another end of the link ($Trans_{Total}$).

$$Link_{ETX} = (Link_{PDR})^{-1} = \frac{Trans_{Total}}{Trans_{Success}} \tag{4}$$

After the path cost calculation, a node has to compute its metric based on the path cost and send it via DIO metric container. In cases where there isn't any metric container in the DIO (It is an option), or the metric container does not carry any metrics, RPL assumes ETX as the default metric. It is worthy to know that instead of additive metrics in RPL, a lexical path selection could be exploited to distinguish paths with the same path cost values; In this kind of situations, the node would select a preferred parent based on a second metric.

3.3. Fuzzy logic objective function

The values of the metrics in all mentioned OFs are either high (1) or low (0) and we were not able to consider a value between 0 or 1 in the OF metrics until the OFFL was introduced [22]. OFFL utilizes another family of logic known as the Fuzzy Logic in which the truth value of a variable could range in the interval [0, 1]. In fuzzy logic theory, linguistic variables are used instead of numeric values to simplify the output expressions (For instance the latency could be low, normal or high, and thus the latency could be referred as a linguistic variable in this definition).

Accordingly, using fuzzy logic in OFFL helps to get more precise link estimations against the default OFs implemented in the standard RPL [22]. The innovative OFFL, combines three link metrics (i.e., end-to-end delay, ETX, and hop count) and one node metric (the node's remaining battery level), to establish the parent selection procedure. The core operations in OFFL could be broadly classified into two categories:

1) **Fuzzification**: The process of dividing a numeric interval into several sub-sets and determining weather a linguistic variable belongs to which of the sub-sets. For instance, the battery level is a linguistic variable, which its value could be mapped into three sub-sets (e.g., low, average, and high). The degree of the linguistic variable's membership in one of the sub-sets is determined by the membership functions. Based on the number

of linguistic variables and their specified values through the membership functions, a number of rules would be employed to generate a dedicated fuzzy output (is also a linguistic variable) according to the combination of composed metrics. The fuzzy output determines the quality of the candidate paths and thus could be used to perform the parent selection.

2) **Defuzzification**: This process is simply the conversion of all of the membership function results related to the linguistic fuzzy output variable, into a numeric value to trigger an action. This numeric value is also known as the crisp value in fuzzy based systems. According to Eq. (5), the Centroid defuzzification method has been employed to obtain the final crisp value [22]. In this regard, to compute the numeric value for the output through this method, three parameters are required: 1) Number of triggered rules based on the determined values by the membership functions in which the strict high or low variables will not be accounted (r), 2) Numeric value related to the outputs of the rules (NV_r), and 3) Membership function value for each of the fuzzy outputs related to each of the mentioned rules ($\gamma(NV_r)$).

$$Output = \frac{\sum_{i=1}^{r} NV_i \times \gamma(NV_i)}{\sum_{i=1}^{r} \gamma(NV_i)} \quad (5)$$

4. The effects of OFs on the IoT systems characteristics

IoT based systems are composed of resource constraint embedded devices. Due to the impacts of communication protocols on these limited resources, it is highly critical to employ the most optimized communication protocols including routing protocols to enhance the system's efficiency from different perspectives. In this regard, OFs play an important role in providing such efficiency. Since the infrastructure of IoT applications is not limited to static nodes, mobility support is a key requirement for many emerging IoT applications, e.g., the location sensing and sharing of location information in smart vehicles and commodity tracking. Unfortunately, the standard RPL is not well suited for mobile IoT topologies and the lack of design in providing OFs with mobility considerations (such as handover and network stability), is one of the key factors for this disproportion. Hence, it is essential to compare and evaluate the state of the art OFs in fixed-node and mobile-node scenarios, not only for determination of their influence on the primitive properties of IoT infrastructures, but also to detect their weakness in mobile applications to develop mobility supporting OFs. Accordingly, in this paper we have evaluated OFs impacts on the key specifications of both, fixed-node and mobile topologies. In this regard, a set of comprehensive experiments have been conducted to evaluate the effects of the most well-known OFs, i.e., OF0, MRHOF, and OFFL from different perspectives including power consumption, E2E delay, reliability and the amount of generated control overhead to maintain the DODAG. In the following, we will describe how these parameters could be affected by the OFs.

4.1. Power consumption

The resource constraint nature of IoT systems, requires a severe consideration for exploiting technologies with lowest possible power consumption. Due to the inaccessibility or portability of the deployed nodes in many IoT applications, the power consumption should be considered carefully due to its indispensable impact on the life time of the entire network. There are many power hungry operating functionalities at different layers of IoT infrastructure (i.e., perception, network, and application layers), which are responsible for the network's entire power consumption. In the meantime, OFs in the structure of RPL (which is part of the network layer) play an important role in selecting those of nodes with higher amounts of remaining energy as the preferred parent to prevent energy hole problem and disconnection of parts of the network.

4.2. End-to-End delay

There are many IoT applications such as Remote Patient Health Monitoring Systems, which require high performance in terms of low response time [51,52]. Accordingly they could be grouped as the real-time IoT applications, where E2E delay is the major concern. The E2E delay parameter represents the required time to transmit a packet from the source to the destination. In a wide group of LLN infrastructures, a set of sensing nodes transmit their data packets towards a sink node in order to deliver them to a remote server (e.g., cloud) through the Internet. Such a transaction creates the MP2P traffic pattern in IoT systems [53]. In this regard, the dominant part of the E2E delay would be the delay between the end nodes and the sink. On the other hand, the RPL's behaviour would certainly impact on the E2E delay and in the meantime, the OF has the most influential effect among the other parts of RPL; Because OF determines the general routing policies in RPL and it directly affects the path selection procedure.

4.3. Reliability

Another important concept in IoT communications is the reliable transmission of data packets in the network. Due to the harsh, dynamic, and lossy nature of the IoT environments, these systems are relatively unreliable [54–56]. This would impose high packet loss and excessive re-transmissions into the network and consequently high energy consumption and long E2E delays in the IoT system. Therefore, reliability plays an essential role in the IoT infrastructure. The way routing protocols select their preferred parent has a direct impact on choosing a reliable path to the sink. Accordingly, the OFs would influence on the number of packet drops in the network.

4.4. Control packet overhead

In order to maintain the DODAG routing tree and preserve the ability of packet transmission in an optimized manner, it is essential to disseminate control packets. As it was discussed earlier, the standard RPL exploits four ICMPv6 control messages (i.e., DIO, DIS, DAO, and DAO-Ack) to propagate routing information. Higher amounts of control overhead due to frequent control packet emissions, leads to higher resource consumption (specially from energy and bandwidth perspectives) as well as possibly more number of collisions and thus lower reliability [57]. Hence, it is desirable to keep the number of transmitted control packets as low as possible. In this regard, RPL exploits the trickle timer algorithm to reduce the number of control packets by increasing their transmission interval in an exponential manner in case of stable topologies. Any inconsistencies in the network, e.g., parent switches, loop detection, and global repairs by the sink (which are directly under the influence of OFs) will reset the trickle to its initialization phase and thus more number of control packets will be disseminated.

5. System setup and results

We have used Cooja [23] simulator to evaluate the impacts of OFs on different aspects of IoT systems characteristics. Cooja is a cycle accurate Java-based simulator, which is able to emulate the typical IoT nodes. Cooja is part of Contiki operating system, which has been developed to be used in IoT embedded devices [58]. Our simulations are established on a class of IoT hardwares known as

Table 2
Zolertia One Specifications.

Parameter	Value
Micro-controller unit (MCU)	2nd MSP430 generation
Architecture	16 bit RISC (Upgraded to 20 bits)
Radio module	CC2420
Operating MCU voltage range	1.8 V<V<3.6 V
CC2420 voltage range	2.1 V<V<3.6 V
Operating temperature	−40° C<θ<+85° C
Operating system clock frequency	f<16 MHz
MCU active mode current @ V_{cc} = 3 V	2 mA
MCU low power (standby) mode current	0.5 µA
Off mode current	0.1 µA
Radio transmitting mode current @ 0 dBm	17.4 mA
Radio receiving mode current	18.8 mA
Radio IDLE mode current	426 µA

Table 3
Simulation environment parameters.

Parameter	Value
Sensing area	0.01 Km^2
Number of transmitter nodes	5, 50, 75
Number of sink nodes	1
Communication range	30 m
Interference range	40 m
Transmission power	0 dBm
Mobile nodes velocity	0.5 m/s
Mobility interval	30 s
Transmission interval (TI)	60 s, 24 s, 12 s
UDP payload size	52 Bytes
Simulation duration	3600 s

Zolertia One (Z1) developed by Zolertia®. This platform exploits the low power Texas Instruments® MSP430 microcontroller as its core and employs the Chipcon® CC2420 radio module for its wireless communications. Some of the most important specifications of this platform has been illustrated in Table 2. In the following, first we explain the designed network scenarios, then the results of experiments will be discussed.

5.1. Experimental methodologies

According to Table 3, in order to conduct our experiments, we have designed and implemented different simulation scenarios in an environment with 0.01 Km^2 area (as in [59,60]) with different network densities. In previously published papers in the context, the number of nodes has been varied significantly from 7 to more than 250 nodes; But, according to Kim et al. [61], the average number of deployed nodes was 49.4 in more than 39 published papers before 2017. With respect to this issue, in order to conduct our experiments, we have varied the number of nodes by 5, 50 and 75, which are randomly distributed in the intended area. As it will be discussed later, in case of dense mobile scenarios, 10 nodes have been set as fixed anchor nodes to receive packets from mobile nodes and forward them to their next hop. The network composes of only one LBR as the root of the tree in order to control the regular nodes through the RPL's ICMPv6 control messages. Regular nodes are set to transmit User Datagram Protocol (UDP) packets with 52 Bytes of payload to the sink with variable Transmission Intervals (TI) to generate heterogeneous traffic rates.

As it has been illustrated in Table 3, in order to have a set of complementary evaluations, three levels of TI has been considered to generate different traffic rates, i.e., low (60 s), moderate (24 s), and high (12 s). The Z1 nodes are set to send their data packets with transmission power of 0 dBm. In addition, these nodes could cover an area up to a radius of 30 m with interference range of 40 m. Finally, to have a fair comparison among different scenarios, in each of scenarios, the simulation was lasted for 3600 s considering the network's convergence time.

5.2. Fixed node experimental results

In the first stage, we evaluate the effects of OFs on the power consumption, E2E delay, reliability and the amount of control overhead in a fixed-node IoT topology using the mentioned configurations. In the following, the results of experiments will be discussed.

5.2.1. Power consumption evaluation

The total power consumption of a node in an IoT infrastructure involves four components. The first is the consumed power in a node during the listen phase to receive the arriving packets from other nodes. The second contributor is the consumed power due to the leakage current in the Low Power Mode (LPM). The third component is the consumed power due to CPU processes in the active mode, and finally the last contributor of the power consumption is the consumed power over the transmission phase to send data packets. In order to calculate the power consumption of a node, the Cooja simulator employs the On-Line Node-Level Energy Estimation technique [62] in which the overall energy consumption (E_{total}) is computed based on Eq. (6).

$$E_{total} = V.\left[I_a t_a + I_{lpm} t_{lpm} + I_{tr} t_{tr} + I_{rc} t_{rc} + \sum_{i=1}^{n} I_p t_p \right] \quad (6)$$

where E_{total} is the total amount of consumed energy in the node, V is the platform's supply voltage, I_a is the draining current from the processor in active mode, I_{lpm} is the drained current in low power mode, I_{tr} and I_{rc} are the drained currents due to transmission and reception activities respectively, and I_p corresponds to the current of the peripheral parts, e.g., LEDs and buttons. The t parameter represents the amount of time that each module has spent in a certain phase. Consequently, according to Eq. (7), in order to calculate the amount of consumed power in a node, E_{total} will be divided by the node's on-time which is the sum of CPU's spent time in active and low power modes.

$$Power = \frac{E_{total}}{t_{cpu} + t_{lpm}} \quad (7)$$

Among the power consumption contributors in an IoT node, OFs can considerably affect the activity of the transmitter module due to their impact on the parent and path selection procedures, which could lead into longer radio up-times in a duty-cycled network and more amounts of power consumption. In addition, OFs could increase the CPU power in active mode by imposing more complex calculations for determining the next hop in the routing algorithms. Accordingly, it is important to have a light-weight OF, which not only optimizes the path selection procedure, but it also balances the traffic load among the candidate parents.

Fig. 3 depicts the average amount of power consumption for an IoT node in different network scenarios. Accordingly, increasing the

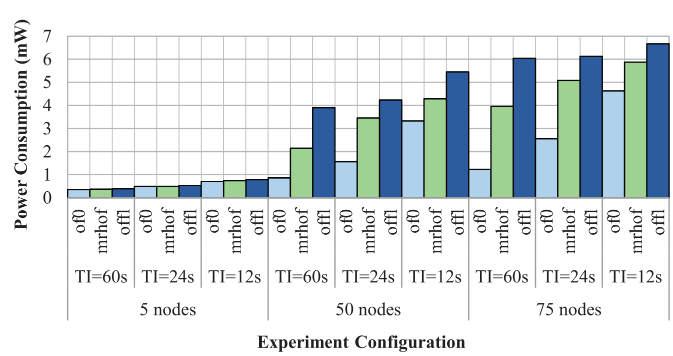

Fig. 3. Average power consumption of a node in different fixed-node IoT network scenarios.

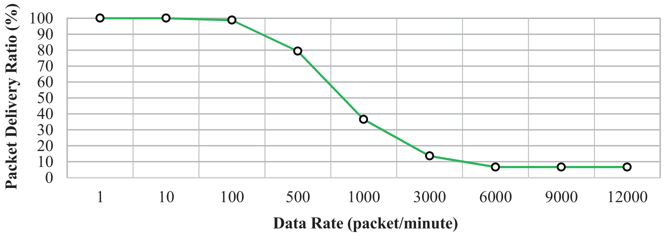

Fig. 4. Effect of Data Rate on PDR.

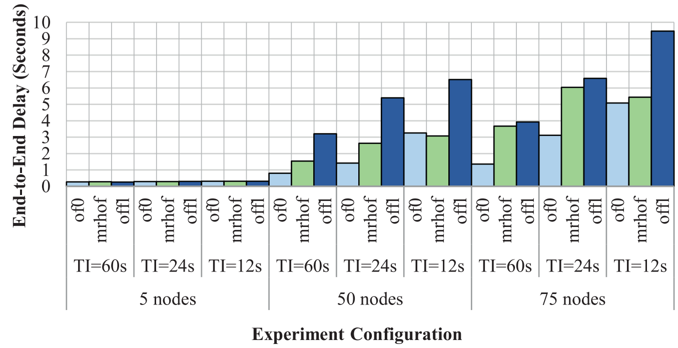

Fig. 5. Average E2E delay in different fixed-node IoT network scenarios.

number of nodes in the network or reducing their data Transmission Interval (TI) would increase their activity due to their communication efforts; Hence, they will consume more amounts of power in the DODAG. As it can be seen in Fig. 3, when the number of distributed nodes in the IoT infrastructure is low (i.e., 5 nodes in our simulations), independent from type of the OF, the consumed power in all of the scenarios are almost the same. In contrast, when the number of nodes in an IoT infrastructure increases, the more complex calculations in OFFL imposes more power consumption to the computational phase in comparison with the other OFs. In addition, since OFFL and MRHOF use ETX metric to guarantee the selection of more reliable links in their communications, they prefer longer and more reliable links instead of shorter and unreliable links (with less number of nodes) towards the sink. This behaviour could involve more number of nodes on the path which leads into increment of the average power consumption per node. Consequently, OFFL and MRHOF have imposed more amounts of power consumption to the nodes compared with OF0. According to Fig. 3, based on the configuration of the network, OFs can affect the average amount of consumed power in each node by up to 80%. This verifies that selecting an appropriate OF based on the application requirements, plays an important role in the power consumption of IoT nodes and their life time.

5.2.2. E2E delay evaluation

To compare the OFs in terms of performance, we have measured the average E2E delay of packet transmissions between sensor nodes and the sink in different network configurations. One of the main contributors to the increment of E2E delay in a path, is the occurrence of congestion which enforces nodes to drop the existing packets in their buffers. According to Fig. 4, by decreasing the TI of a node (increasing the data rate) in a specific path with adequate bandwidth, the PDR will be drastically reduced which indicates that the data rate is a key factor in the concept of congestion. On the other hand, dropping packets in congested scenarios necessitates the retransmission of packets which will directly affect the ETX of the paths. Hence, ETX metric has a critical role in the path selection in case of congested networks.

According to Fig. 5, in situations where only a few number of nodes (5 nodes) with either low or high data rates are in the network, OFs act relatively the same. In order to evaluate the exact behavior of the OFs, we have assumed high Link Success Rates (LSR) in our simulations. Accordingly, in more dense topologies, shorter paths with less number of nodes would not only provide better paths in terms of reliability, but also provide better paths in terms of E2E delay. Hence, as it has been presented in Fig. 5, OF0 could decrease the average amount of latency by up to 63% and 75% compared with MRHOF and OFFL respectively. Based on our observations, by decreasing the transmission interval along with deploying more number of nodes to the infrastructure, the network will be prone to congestion and different paths may provide different ETX values. Accordingly, since MRHOF selects the path with lower ETX, it reduces the number of retransmissions in the nodes. Hence, MRHOF would reduce the amount of E2E delay by up to 53% in comparison with OFFL. On the other hand, since OFFL exploits remaining energy of the nodes, it would prefer slower paths with better energy efficiency and thus it has imposed up to 4.03x and 2.1x of more E2E delay compared with OF0 and MRHOF respectively. According to our observations, OFs' impact on the average E2E delay in the network could be as high as 75%. This clarifies the importance of OFs on the E2E delay in IoT systems.

5.2.3. Reliability evaluation

In order to evaluate and report the reliability of OFs, we measure PDR. To this end we have used Eq. (8), in which the PDR is computed as the ratio of received packets by the sink to the total number of transmitted packets towards the sink in the network.

$$PDR = \frac{ReceivedPacketsbytheSink}{TransmittedPacketstotheSink} \tag{8}$$

According to Fig. 6, independent from type of the OF, by assuming a network with a constant number of nodes, the increment of data rate will degrade the amount of PDR value. In addition, increasing the number of nodes in the IoT infrastructure with a constant TI would also reduce the PDR; Because, more number of nodes in the network will have two consequences: 1) Increment of interference between the nodes, and 2) Generation of more number of data and control packets, which will increase the probability of congestion and consequently the packet drops in the node buffers. Assuming high LSR values in our simulations enables us to evaluate the exact behavior of OFs in terms of PDR. Accordingly, as it has been illustrated in Fig. 6, in a specific network configuration, from MRHOF point of view, all of the existing candidate paths have relatively the same QoS in terms of ETX; Because the path selection in MRHOF is based on the number of retransmissions, which is directly affected by LSR. On the other hand, since OF0 selects the path with the minimum number of hops to the root, shorter paths will provide lower probability of packet drops due to their high LSR and fewer number of hops. Accordingly, OF0 has improved the PDR by up to 33% compared with MRHOF. It is

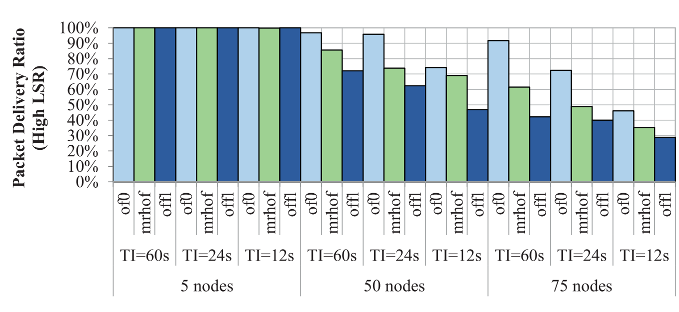

Fig. 6. Average PDR in different fixed-node IoT network scenarios.

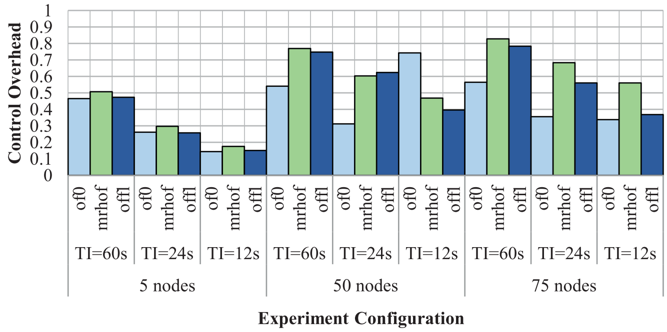

Fig. 7. Average Control Packet Overhead in different fixed-node IoT network scenarios.

worthy to mention that benefit of selecting MRHOF as the OF in an RPL based IoT infrastructure is more obvious when the LSR is low [63]. On the other hand since OFFL exploits two other metrics than ETX and hop count in its structure, it could not compete with MRHOF and OF0 in terms of PDR and according to Fig. 6, it has reduced the amount of PDR by up to 54% and 7% compared with OF0 and MRHOF respectively.

Based on our evaluations, low data rates and limited number of deployed nodes will create a situation where the furthest nodes from the sink, do not have a considerable distance with it, and there are only a few number of hops ahead of them towards the sink. Hence, the PDR in different paths would be high and similar, and its value does not alter by employing different OFs. Generally, based on our observations, OFs have affected the value of PDR by at most 54%, which is also a sign for reliability experts to pay more attention to OFs in the concept of reliability, specially in dense IoT systems.

5.2.4. Control packet overhead evaluation

In order to measure the amount of generated control packet overhead in each node, we have calculated the ratio of control packets to the total number of generated packets in the nodes. As it has been depicted in Fig. 7, independent from type of the OF, by considering a topology with a fixed number of nodes, the increment of data rate (reducing TI) would reduce the control packet overhead; Because by increasing the data rate, the rate of control packets will not be affected as much as data packets. Hence, the overhead will be decreased. On the other hand, by assuming constant data rates, by adding more number of nodes to the network, the amount of control packet overhead will be increased. This is due to more amounts of fluctuations in the quality of the links from different perspectives in case of higher number of nodes. Thus, RPL nodes need to propagate higher amounts of control packets for maintaining the DODAG.

In the meantime, according to our observations, OFs structure has a direct impact on the control packet overhead. As it has been illustrated in Fig. 7, MRHOF and OFFL have imposed more amounts of control packet overhead to the network in comparison with OF0; Because they use metrics for their path selection procedure while OF0 does not exploit any metrics and it only selects the paths based on the number of hops to the root. Accordingly, since QoS of the paths in terms of metrics are time-varying, the probability of switching paths by MRHOF and OFFL is higher and they would impose more control packets to the network. Unlikely, since OF0 does not use any metrics and its path selection is based on the number of hops, there will be no parent switches in the network while there is not any link failures. Thus, according to Fig. 7, OF0 could reduce the amount of control packet overhead by up to 48% and 50% compared with MRHOF and OFFL respectively. Meanwhile, since OFFL has more number of metrics and constraints in its structure against MRHOF, it is more likely for MRHOF to change the paths. Accordingly, OFFL could decrease the control packet overhead by up to 34% compared with MRHOF. Based on our observations, depending on the configuration of the system, OFs could affect the amount of control packet overhead by up to 50%.

5.3. Mobile node experimental results

In mobile IoT infrastructures, alterations in the links are more frequent. Thus, there is a high demand for designing OFs, which could be adapted to these fluctuations. But, in advance we need to highlight the deficiencies of the current OFs in mobile conditions. Hence, in this section we have evaluated the functionality of the OFs in mobile scenarios. In this regard we need to adopt a mobility model for our evaluations. So far, Various mobility models such as random waypoint [64], random walk [65], Self-similar Least Action Walk (SLAW) [66], and Reference Point Group Mobility (RPGM) [67] have been presented to be exploited in mobile applications based on their characteristics. Among the different mobility models, those which are based on random patterns could be more beneficial for evaluating the OF's behaviour in mobile infrastructures. Because the random specification of a mobility model could provide the worst case scenario for OFs and thus, we would be able to guarantee their better efficiency in case of more predictable mobility models.

In this regard, we have proposed and employed a modified version of random walk mobility model with angle view (RWAV), which is able to consider angles of movement in a bounded area. According to Algorithm 1, in our proposed model, nodes traverse a direction in a specified interval, in which their angle of movement varies smoothly. More precisely, as it has been depicted in Fig. 8, intervals are built upon several sub-intervals in which at the end of each sub-interval, nodes select a new angle from a constraint range (in our simulations, from a 30° range). At the end of the intervals, every node randomly selects a new direction with a completely new angle and a new interval begins. In case of reaching the boundaries, nodes will change their direction immediately in

Algorithm 1: Random Walk Mobility Model with Angle View (RWAV).

```
Input: Node's time in Seconds, Node's velocity, Area boundaries, Interval
Output: Update node coordinates and generate mobility trace
/* The following code is running on each node                          */
if ((Node.Time mod Node.Interval = 0)
or(Node.X=Node.XMinBoundry) or(Node.X=Mode.XMaxBoundry)
or(Node.Y=Node.YMinBoundry) or(Node.Y=Node.YMaxBoundry)) then
 |  Node.θ ← Random[∡ 0°, ∡ 360°];
else
 |  Node.θ ← Node.θ + Random[∡ -15°, ∡ 15°];
end
Node.X ← Node.X + Node.Velocity * cos(Node.θ);
Node.Y ← Node.Y + Node.Velocity * sin(Node.θ);
/* Now the code evaluates the area constraints for node coordinates;
   Because they should not exceed the area boundaries.                 */
if (Node.X <Node.XMinBoundary) then
 |  Node.X ← Node.XMinBoundary;
else
 |  if (Node.X >Node.XMaxBoundary) then
 |   |  Node.X ← Node.XMaxBoundary;
 |  end
end
if (Node.Y <Node.YMinBoundary) then
 |  Node.Y ← Node.YMinBoundary;
else
 |  if (Node.Y >Node.YMaxBoundary) then
 |   |  Node.Y ← Node.YMaxBoundary;
 |  end
end
```

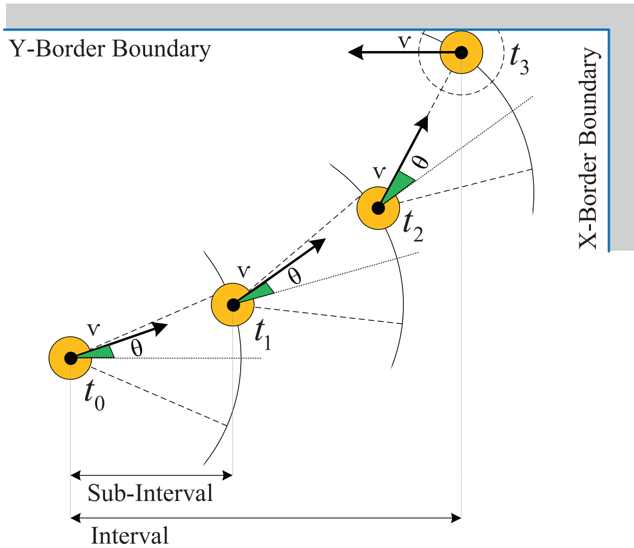

Fig. 8. Structure of node's movement in the modified Random Walk Mobility Model with Angle View (RWAV).

order to move back towards the specified area. In our evaluations, the time intervals and the node's velocity have been set to constant values (Table 3). It is worthy to mention that the standard Cooja does not support mobility itself. Therefore we have added a mobility plug-in to Cooja to be able to feed the generated mobility trace file into it.

5.3.1. Power consumption evaluation

According to Fig. 9, the behaviour of the nodes in terms of their average power consumption in mobile IoT infrastructures is similar to what we have seen in fixed node scenarios. In this regard, adopting more number of nodes in the system or increasing their data rate would lead into increment of the average power consumption of the nodes. Nevertheless, the average amount of consumed power in mobile infrastructures is higher than fixed node systems. In the meantime, OFFL could impose up to 3.3x and 1.2x of more power consumption to the nodes in comparison with OF0 and MRHOF respectively. This is due to more computation complexities for path selection in low density topologies and more number of exploited metrics in the structure of OFFL, which prevents an optimized path selection in terms of power consumption in more dense networks. On the other hand, the OF0 chooses the shortest path in terms of hop count and thus, fewer number of nodes will be involved in the routing procedure. But, MRHOF is more interested in selecting longer and more reliable paths instead

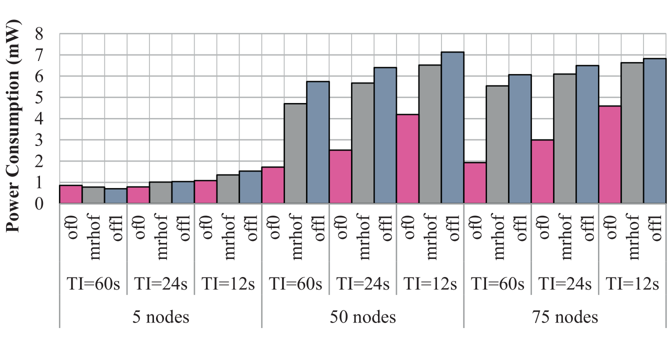

Fig. 9. Average power consumption of a node in different mobile IoT network scenarios.

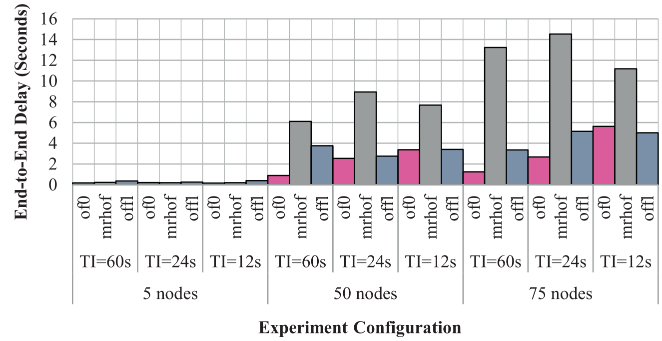

Fig. 10. Average E2E delay in different mobile IoT network scenarios.

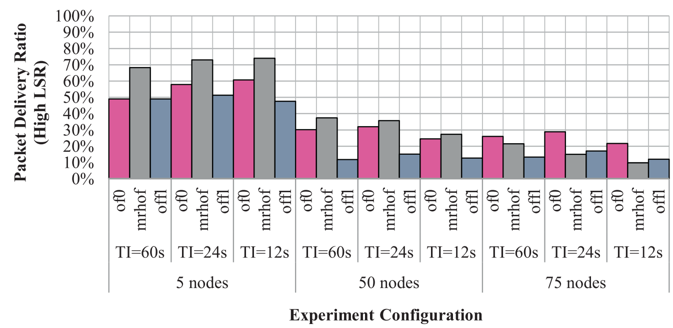

Fig. 11. Average PDR in different mobile IoT network scenarios.

of shorter and unreliable ones. Hence, in case of using MRHOF, more number of nodes will be get involved and their activity will be increased, which leads into more amounts of power consumption. Accordingly, OF0 could reduce the average power consumption of the nodes by up to 65% compared with MRHOF. Based on our observations, in mobile IoT infrastructures, OFs could impact on the average power consumption by up to 70%.

5.3.2. E2E delay evaluation

In case of mobile IoT applications, the distance between two nodes will be time-varying. Hence, the distance between nodes and the sink will be either increasing or decreasing. Accordingly, the use of ETX enforces MRHOF to stick with the route which provides higher reliability (Fig. 11), even if it takes more time to deliver data packets to the destination. Consequently, as long as the received signal is above the threshold or there is not any failures on the links of the path, MRHOF insists on the reliable path. Thus, according to Fig. 10, MRHOF could impose up to 91% and 75% of more E2E delay compared with OF0 and OFFL respectively.

In order to evaluate the behaviour of OF0 and OFFL, the two following situations should be analyzed: 1) Low data rates, and 2) High data rates. According to Fig. 10, in case of low data rates, independent from number of deployed nodes, the OFFL has imposed more amounts of latency to the network; Because in addition to the hop count and delay metrics, it considers the remaining energy level of the nodes which prevents its optimized path selection in terms of E2E delay. In this regard, based on our observations, OFFL has imposed up to 4.2x of more E2E delay compared with OF0. On the other hand, as it has been depicted in Fig. 10, by increasing the data rates, OF0 and OFFL has acted similarly. In general, in case of mobile IoT applications, OFs could increase the amount of E2E delay by more than 91% which could be a part of scholars considerations in designing appropriate OFs for mobile IoT infrastructures.

5.3.3. Reliability evaluation

The movement of the nodes in mobile IoT applications causes significant alterations in the network, e.g., nodes distances from one another, the relative Received Signal Strength Index (RSSI), neighbour and parent tables, preferred parent of the nodes and many more inconsistencies. Accordingly, the successful receipt of the packets at the destination would be more complicated compared with fixed node topologies. In this regard, based on the provided information in Figs. 6 and 11, the average amount of PDR in mobile conditions has been degraded by 55% compared with fixed node scenarios. On the other hand, deploying more number of nodes in the network and increasing their data rates will also reduce the PDR values.

In the meantime, in case of low density scenarios, the distance between the nodes is relatively longer than their distance in more dense topologies. Hence, even with high values of LSR, the quality of the links are continuously fluctuating. In this regard, since MRHOF exploits ETX in its structure, it could adjust itself with such conditions and it could optimize the path selection in terms of reliability. As it has been depicted in Fig. 11, MRHOF could enhance the average amount of PDR by up to 28% and 69% compared with OF0 and OFFL respectively. But in case of dense topologies, the distance of the nodes would be less and the links will be shorter. Thus, the parent switches would not impose significant changes to the path quality (in terms of reliability) compared with situations where only a few number of mobile nodes are moving in the area. Accordingly, by assuming high values of LSR in the network, the situation would be similar to what we had in fixed node scenarios where the OF0 had improved the PDR. Based on our observations, OF0 could improve the average amount of PDR in the mobile IoT applications by up to 55% and 49% compared with MRHOF and OFFL respectively. In summary, in mobile IoT infrastructures, OFs could have impacts by up to 69% on the PDR which requires certain considerations.

5.3.4. Control packet overhead evaluation

By comparing the provided information in Figs. 7 and 12, in case of mobile IoT applications, independent from type of the OF, the overall control overhead due to control packets has been increased compared with fixed node scenarios. This is due to more frequent alterations in the characteristics of the DODAG, which necessitates the dissemination of control packets in the network. According to Fig. 12, similar to what we have had in the fixed-node scenarios, in a topology with a constant number of nodes, reducing the TI (increasing the data rate) would decrease the amount of control packet overhead. Also, by assuming constant data rates, adding more number of nodes to the network, will increase the amount of control packet overhead.

Meanwhile, similar to the fixed node scenarios, as it has been presented in Fig. 12, MRHOF and OFFL have generated more amounts of control packets in the network compared with OF0; Because they employ metrics for their path selection while OF0 does not exploit any metrics and it only selects the paths based on the number of hops to the root. Accordingly, since QoS of the paths in terms of metrics are time-varying (which will be more intense due to node's movements in mobile infrastructures), the probability of switching the paths by MRHOF and OFFL is higher and thus, they will impose more control packets to the network. Unlikely, since OF0 does not use any metrics in its path selection, there will be no parent switches in the network while there is not any link failures. Thus, according to Fig. 12, OF0 could reduce the amount of control packet overhead by up to 58% and 30% compared with MRHOF and OFFL respectively. In the meantime, since OFFL has more number of metrics and constraints in its structure against MRHOF, there is a higher chance for MRHOF to change the paths. Accordingly, OFFL could decrease the control packet overhead by up to 44% compared with MRHOF. In summary, in case of having a mobile IoT infrastructure, depending on the configuration of the system, OFs could affect the amount of control packet overhead by up to 58%.

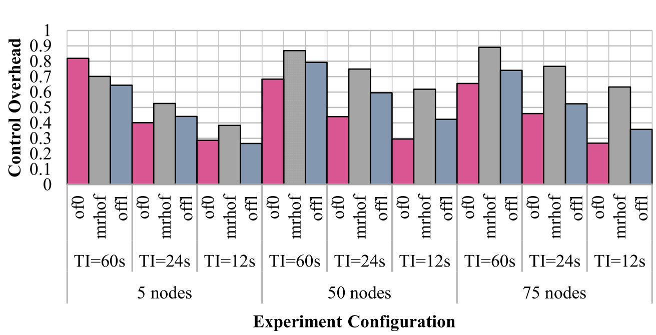

Fig. 12. Average Control Packet Overhead in different mobile IoT network scenarios.

6. Conclusion

Routing is a major concern in IoT infrastructures due to its direct impact on the communications of the existing embedded devices. In this regard, RPL was introduced to comply with the primitive requirements of static IoT applications through OFs. Nevertheless, since RPL was not designed to satisfy mobile IoT applications' requirements, its response to topology alterations is weak and expensive. The lack of appropriate OFs for mobile applications is one the most important reasons for such disproportion. Accordingly, the functionality of the existing OFs should be carefully studied in order to determine their challenges for future evaluations and OF designs. In this regard, this paper has evaluated the effect of OFs on the primary requirements of IoT infrastructures in both, mobile and static topologies. Our evaluations, which were conducted on Cooja, have shown that in fixed-node IoT infrastructures, OFs could affect the power consumption by up to 80%, while their impact on reliability and E2E delay would be as high as 54% and 75%, respectively. Further observations have proven that mobility would reduce OFs effect on the power consumption by more than 10% in comparison with static IoT infrastructures, while it can increase their influence on reliability and E2E delay by nearly 15% and 16%, respectively.

Declaration of Competing Interest

We would like to kindly inform the reader that we don't have any conflict of interests.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.micpro.2019.05.010.

References

[1] B. Safaei, A.M.H. Monazzah, M.B. Bafroei, A. Ejlali, Reliability side-effects in internet of things application layer protocols, in: Proceedings of the 2nd International Conference on System Reliability and Safety (ICSRS), IEEE, 2017, pp. 207–212.

[2] J. Manyika, M. Chui, P. Bisson, J. Woetzel, R. Dobbs, J. Bughin, D. Aharon, The internet of things: mapping the value beyond the hype, in: McKinsey Global Institute, McKinsey, 2015, p. 3.

[3] D. Christin, A. Reinhardt, P.S. Mogre, R. Steinmetz, et al., Wireless sensor networks and the internet of things: selected challenges, Proceedings of the 8th GI/ITG KuVS Fachgespräch Drahtlose sensornetze (2009) 31–34.

[4] L. Mainetti, L. Patrono, A. Vilei, Evolution of wireless sensor networks towards the internet of things: a survey, in: SoftCOM 2011, 19th International Conference on Software, Telecommunications and Computer Networks, IEEE, 2011, pp. 1–6.

[5] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of Things (IoT): a vision, architectural elements, and future directions, Future Gener. Comput. Syst. 29 (7) (2013) 1645–1660.

[6] C. Alcaraz, P. Najera, J. Lopez, R. Roman, Wireless sensor networks and the internet of things: do we need a complete integration? 1st International Workshop on the Security of the Internet of Things (SecIoT10), 2010.
[7] M. Dohler, T. Watteyne, T. Winter, D. Barthel, Routing requirements for urban low-power and lossy networks, Technical Report, 2009.
[8] K. Pister, P. Thubert, S. Dwars, T. Phinney, Industrial routing requirements in low-power and lossy networks, Technical Report, 2009.
[9] A. Brandt, G. Porcu, Home automation routing requirements in low power and lossy networks (2010).
[10] J. Martocci, W. Vermeylen, N. Riou, P.D. Mil, Building automation routing requirements in low power and lossy networks (2010).
[11] P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, R. Alexander, Rpl: ipv6 routing protocol for low power and lossy networks, RFC 6550 (2012).
[12] C.R. Mesh, Cisco Resilient Mesh, 2018 [Online]. Available: https://www.cisco.com/ Accessed: [Mar. 2, 2019].
[13] J.-P. Vasseur, M. Kim, K. Pister, N. Dejean, D. Barthel, Routing metrics used for path calculation in low-power and lossy networks, Technical Report, 2012.
[14] P. Thubert, Objective function zero for the routing protocol for low-power and lossy networks (rpl) (2012).
[15] O. Gnawali, The minimum rank with hysteresis objective function (2012).
[16] E. Ancillotti, R. Bruno, M. Conti, Reliable data delivery with the ietf routing protocol for low-power and lossy networks, IEEE Trans. Ind. Inform. 10 (3) (2014) 1864–1877.
[17] S. Capone, R. Brama, N. Accettura, D. Striccoli, G. Boggia, An energy efficient and reliable composite metric for rpl organized networks, in: Embedded and Ubiquitous Computing (EUC), 2014 12th IEEE International Conference on, IEEE, 2014, pp. 178–184.
[18] Y. Tahir, S. Yang, J. McCann, Brpl: backpressure rpl for high-throughput and mobile iots, IEEE Trans. Mobile Comput. (2017).
[19] M. Barcelo, A. Correa, J.L. Vicario, A. Morell, X. Vilajosana, Addressing mobility in rpl with position assisted metrics, IEEE Sensors J. 16 (7) (2016) 2151–2161.
[20] H. Fotouhi, D. Moreira, M. Alves, P.M. Yomsi, Mrpl+: a mobility management framework in rpl/6lowpan, Comput. Commun. 104 (2017) 34–54.
[21] O. Gaddour, A. Koubâa, R. Rangarajan, O. Cheikhrouhou, E. Tovar, M. Abid, Co-rpl: Rpl routing for mobile low power wireless sensor networks using corona mechanism, in: Industrial Embedded Systems (SIES), 2014 9th IEEE International Symposium on, IEEE, 2014, pp. 200–209.
[22] O. Gaddour, A. Koubâa, M. Abid, Quality-of-service aware routing for static and mobile ipv6-based low-power and lossy sensor networks using rpl, Ad Hoc Netw. 33 (2015) 233–256.
[23] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, T. Voigt, Cross-level sensor network simulation with cooja, in: Local computer networks, proceedings 2006 31st IEEE conference on, IEEE, 2006, pp. 641–648.
[24] B. Safaei, A.M.H. Monazzah, T. Shahroodi, A. Ejlali, Objective function: a key contributor in internet of things primitive properties, in: 2018 Real-Time and Embedded Systems and Technologies (RTEST), IEEE, 2018, pp. 39–46.
[25] N.A. Pantazis, S.A. Nikolidakis, D.D. Vergados, Energy-efficient routing protocols in wireless sensor networks: a survey, IEEE Commun. Surv. Tutor. 15 (2) (2013) 551–591.
[26] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, P. Levis, Collection tree protocol, in: Proceedings of the 7th ACM conference on embedded networked sensor systems, ACM, 2009, pp. 1–14.
[27] S. Dawson-Haggerty, A. Tavakoli, D. Culler, Hydro: a hybrid routing protocol for low-power and lossy networks, in: Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on, IEEE, 2010, pp. 268–273.
[28] H.-f. Xie, F. Zeng, G.-q. Zhang, D.-l. Su, Simulation research on routing protocols in zigbee network, in: Proceedings of the 6th International Asia Conference on Industrial Engineering and Management Innovation, Springer, 2016, pp. 891–898.
[29] C. Perkins, E. Belding-Royer, S. Das, Ad hoc on-demand distance vector (AODV) routing, Technical Report, 2003.
[30] R. Fonseca, S. Ratnasamy, J. Zhao, C.T. Ee, D. Culler, S. Shenker, I. Stoica, Beacon vector routing: scalable point-to-point routing in wireless sensornets, in: Proceedings of the 2nd Conference on Symposium on Networked Systems Design & Implementation-Volume 2, USENIX Association, 2005, pp. 329–342.
[31] LoRa, Lora-alliance, 2017 [Online]. Available: https://www.loraalliance.org, Accessed: [Mar. 2, 2019].
[32] Sigfox, sigfox: The Global Communications Service Provider for the Internet of Things (IoT), 2017 [Online]. Available: https://www.sigfox.com/en, Accessed: [Mar. 2, 2019].
[33] B. Sartori, S. Thielemans, M. Bezunartea, A. Braeken, K. Steenhaut, Enabling rpl multihop communications based on lora, in: 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), IEEE, 2017, pp. 1–8.
[34] X. Liu, Z. Sheng, C. Yin, F. Ali, D. Roggen, Performance analysis of routing protocol for low power and lossy networks (rpl) in large scale networks, IEEE IoT J. (2017).
[35] M. Banh, et al., Energy balancing rpl-based routing for internet of things, in: Proceedings of the 6th IEEE International Conference on Communications and Electronics (ICCE), IEEE, 2016, pp. 125–130.
[36] N. Sousa, et al., Eraof: a new rpl protocol objective function for internet of things applications, in: proceedings of the 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech), IEEE, 2017, pp. 1–5.
[37] R. Ullah, et al., Energy and congestion-aware routing metric for smart grid ami networks in smart city, IEEE Access 5 (2017) 13799–13810.
[38] L. Lassouaoui, et al., Evaluation of energy aware routing metrics for rpl, in: Proceedings of the 12th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), IEEE, 2016, pp. 1–8.
[39] J. Nurmio, et al., Equalizing energy distribution in sensor nodes through optimization of rpl, in: Proceedings of the IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), IEEE, 2015, pp. 83–91.
[40] S. Capone, et al., An energy efficient and reliable composite metric for rpl organized networks, in: Proceedings of the 12th IEEE International Conference on Embedded and Ubiquitous Computing (EUC), IEEE, 2014, pp. 178–184.
[41] N.M. Shakya, et al., Seeof: smart energy efficient objective function: adapting rpl objective function to enable an ipv6 meshed topology solution for battery operated smart meters, in: Proceedings of the Global Internet of Things Summit (GIoTS), IEEE, 2017, pp. 1–6.
[42] M. Qasem, et al., A new efficient objective function for routing in internet of things paradigm, in: Proceedings of the IEEE International Conference on Standards for Communications and Networking (CSCN), IEEE, 2016, pp. 1–6.
[43] O. Iova, et al., Improving the network lifetime with energy-balancing routing: application to rpl, in: Proceedings of the 7th IFIP Wireless and Mobile Networking Conference (WMNC), IEEE, 2014, pp. 1–8.
[44] O. Iova, et al., Using multiparent routing in rpl to increase the stability and the lifetime of the network, Ad Hoc Netw. 29 (2015) 45–62.
[45] X. Yang, et al., Stability metric based routing protocol for low-power and lossy networks, in: Proceedings of the IEEE International Conference on Communications (ICC), IEEE, 2014, pp. 3688–3693.
[46] M. Zhao, et al., An energy-efficient region-based rpl routing protocol for low-power and lossy networks, IEEE IoT J. 3 (6) (2016) 1319–1333.
[47] W. Xiao, et al., An optimization of the object function for routing protocol of low-power and lossy networks, in: Proceedings of the 2nd International Conference on Systems and Informatics (ICSAI), IEEE, 2014, pp. 515–519.
[48] J.A. Hatem, et al., Enhancing routing protocol for low power and lossy networks, in: Proceedings of the 13th International Wireless Communications and Mobile Computing Conference (IWCMC), IEEE, 2017, pp. 753–758.
[49] H.-S. Kim, et al., Load balancing under heavy traffic in rpl routing protocol for low power and lossy networks, IEEE Trans. Mobile Comput. 16 (4) (2017) 964–979.
[50] B. Safaei, A.A.M. Salehi, M. Shirbeigi, A.M.H. Monazzah, A. Ejlali, Pedal: power-delay product objective function for internet of things applications, in: Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, ACM, 2019, pp. 892–895.
[51] P. Verma, S.K. Sood, Fog assisted-iot enabled patient health monitoring in smart homes, IEEE IoT J. (2018).
[52] A. Sharma, R. Kumar, An optimal routing scheme for critical healthcare hth servicesan iot perspective, in: Image Information Processing (ICIIP), 2017 Fourth International Conference on, IEEE, 2017, pp. 1–5.
[53] T. Watteyne, M. Palattella, L. Grieco, Using IEEE 802.15. 4e time-slotted channel hopping (TSCH) in the internet of things (IoT): problem statement, Technical Report, 2015.
[54] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, M. Ayyash, Internet of things: a survey on enabling technologies, protocols, and applications, IEEE Commun. Surv. Tutor. 17 (4) (2015) 2347–2376.
[55] J. Quevedo, M. Antunes, D. Corujo, D. Gomes, R.L. Aguiar, On the application of contextual iot service discovery in information centric networks, Comput. Commun. 89 (2016) 117–127.
[56] P. Lv, Z. Cai, J. Xu, M. Xu, Multicast service-oriented virtual network embedding in wireless mesh networks, IEEE Commun. Lett. 16 (3) (2012) 375–377.
[57] T. Clausen, U. Herberg, M. Philipp, A critical evaluation of the ipv6 routing protocol for low power and lossy networks (rpl), in: Wireless and Mobile Computing, Networking and Communications (WiMob), 2011 IEEE 7th International Conference on, IEEE, 2011, pp. 365–372.
[58] A. Dunkels, B. Gronvall, T. Voigt, Contiki-a lightweight and flexible operating system for tiny networked sensors, in: Local Computer Networks, 2004. 29th Annual IEEE International Conference on, IEEE, 2004, pp. 455–462.
[59] W. Khallef, M. Molnar, A. Benslimane, S. Durand, Multiple constrained qos routing with rpl, in: 2017 IEEE International Conference on Communications (ICC), IEEE, 2017, pp. 1–6.
[60] A. Kalmar, R. Vida, M. Maliosz, Caesar: a context-aware addressing and routing scheme for rpl networks, in: 2015 IEEE International Conference on Communications (ICC), IEEE, 2015, pp. 635–641.
[61] H.-S. Kim, J. Ko, D.E. Culler, J. Paek, Challenging the ipv6 routing protocol for low-power and lossy networks (rpl): a survey, IEEE Commun. Surv. Tutor. 19 (4) (2017) 2502–2525.
[62] A. Dunkels, F. Osterlind, N. Tsiftes, Z. He, Software-based on-line energy estimation for sensor nodes, in: Proceedings of the 4th workshop on Embedded networked sensors, ACM, 2007, pp. 28–32.
[63] M. Zhao, A. Kumar, P.H.J. Chong, R. Lu, A comprehensive study of rpl and p2p-rpl routing protocols: implementation, challenges and opportunities, Peer-to-Peer Netw. Appl. 10 (5) (2017) 1232–1256.
[64] D.B. Johnson, D.A. Maltz, Dynamic source routing in ad hoc wireless networks, in: Mobile computing, Springer, 1996, pp. 153–181.
[65] T. Camp, J. Boleng, V. Davies, A survey of mobility models for ad hoc network research, Wirel. Commun. Mobile Comput. 2 (5) (2002) 483–502.
[66] H. Lamaazi, N. Benamar, M.I. Imaduddin, A. Habbal, A.J. Jara, Mobility support for the routing protocol in low power and lossy networks, in: 2016 30th

International Conference on Advanced Information Networking and Applications Workshops (WAINA), IEEE, 2016, pp. 809–814.
[67] X. Hong, M. Gerla, G. Pei, C.-C. Chiang, A group mobility model for ad hoc wireless networks, in: Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems, ACM, 1999, pp. 53–60.

Bardia Safaei (IEEE Student Member Since 2013) received his B.Sc. degree in computer engineering from K. N. Toosi university of technology, Tehran, Iran, in 2014 and the M.Sc degree in computer engineering from Sharif University of technology, Tehran, Iran, in 2016 respectively. Currently he is pursuing the Ph.D. degree in computer engineering at Sharif university of technology. He was a member of Dependable Systems Laboratory (DSL) during 2014 to 2017 and currently he is a member of Embedded Systems Research Laboratory (ESRLab). He is honored to be selected as a member of national elites foundation since 2016. His research interests include power efficiency and dependability challenges in Internet of Things (IoT), Cyber Physical Systems (CPS), Embedded Systems, Distributed Storage Systems and Cloud Computing.

Ali Asghar Mohammad-Salehi received his B.E. degree in computer science and engineering from Sharif University of Technology (SUT), Tehran, Iran in 2017. He is currently pursuing the M.E. degree in computer architecture engineering at Sharif University of Technology. He is a fully expert of C, C++, Python and Javascript programming languages and he has three years of work experience at reputed computer corporations. He is currently a member of Embedded Systems Research Laboratory (ESRLab). His main research interests include energy efficiency and reliability in Internet of things, embedded systems, wireless sensor networks, and System on Chip (SoC).

Amir Mahdi Hosseini Monazzah received the B.Sc. degree in computer engineering from Islamic Azad University (South Tehran branch), Tehran, Iran, in 2009, and the M.Sc. and Ph.D degree in computer engineering from the Sharif University of Technology, Tehran, Iran, in 2012 and 2017, respectively. As a postdoc fellow he is currently with the school of computer science, institute for research in fundamental sciences (IPM), Tehran, Iran. He has been a member of the Dependable Systems Laboratory since 2010. As a Visiting Researcher, he was with the Embedded Systems Laboratory, University of California, Irvine, CA, USA from 2016 to 2017. His research interests include investigating the reliability challenges of emerging non-volatile memories, fault-tolerant hybrid memory hierarchy design, designing reliable software for unreliable hardware, designing dependable embedded systems, and reliability challenges in multi-core systems.

Alireza Ejlali received the Ph.D. degree in computer engineering from Sharif University of Technology in, Tehran, Iran, in 2006. He is currently an associate professor of computer engineering at Sharif University of Technology. From 2005 to 2006, he was a visiting researcher in the Electronic Systems Design Group, University of Southampton, Southampton, United Kingdom. In 2006, he joined Sharif University of Technology as a faculty member in the department of computer engineering and from 2011 to 2015 he was the director of Computer Architecture Group in this department. His research interests include low power design, real-time embedded systems, and fault-tolerant embedded systems.