IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 29, 2018, accepted February 27, 2018, date of publication April 9, 2018, date of current version May 16, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2817128

CLRPL: Context-Aware and Load Balancing RPL for Iot Networks Under Heavy and Highly Dynamic Load

SEYEDREZA TAGHIZADEH[1], HOSSEIN BOBARSHAD[1], AND HALIMA ELBIAZE[2]
[1]Department of New Sciences and Technologies, University of Tehran, Tehran 141556619, Iran
[2]Department of Computer Science, Université du Québec à Montréal, Montréal, QC H2X 3Y7, Canada

Corresponding author: Hossein Bobarshad (hossein.bobarshad@ut.ac.ir)

ABSTRACT The IPv6 routing protocol for low power and lossy networks (RPL) was accepted as the standard routing protocol for Internet of Things (IoT) by Internet engineering task force on March 2012. Since then, it has been used in different IoT applications. Though RPL considerably deals with IoT network requirements, there are still some open problems to solve, since it was not initially designed for IoT applications. In this paper, we address the problem of packet loss and power depletion in an RPL-based network under heavy and highly dynamic load. We address this problem in three steps: First, we present a context-aware objective function (CAOF), which computes the rank, considering the context of the node. CAOF also avoids thundering herd phenomenon by gradual shifting from a high rank value toward the real rank value. Second, we present a new routing metric, known as context-aware routing metric (CARF) which considers the status of remaining power and queue utilization of parent chain toward the root in a recursive manner while lessening the effect of upstream parents as it gets farther down the path. Through comprehensive evaluations, we show that this metric leads to a better decision about the proper parent in a network with high traffic dynamicity, rather than deciding merely based on the parent rank. Third, we present a new parent selection mechanism, which selects the best parent based on CARF and some other metrics while avoiding routing loops by a simple yet effective countermeasure. Evaluation results reveal improvements in network lifetime while decreasing packet loss in comparison with standard specification of RPL.

INDEX TERMS Context-aware routing, Internet of Things, heavy load, network lifetime, packet loss, dynamic load.

I. INTRODUCTION

In recent years, the Internet of Things (IoT) has grown to be a hot topic among researchers across the world. it has made its way into different scopes like transportation, agriculture, industry, and healthcare due to its features like working in an IP based network and being able of holding thousands or millions of nodes [1], [2]. These nodes are capable of communicating and cooperating with each other to achieve a goal. In regard to the inherent attributes of IoT, like Being IP-Based, large scale, and universally addressable, the Internet Engineering Task Force (IETF) has made some standardization efforts specific to IoT [3]. One of the pertinent IETF working groups is Routing Over Low power and Lossy networks (ROLL). As the name implies, the main objective of this group is to focus on the routing of Low power and Lossy networks (LLN) like IoT. In an LLN, there are some power constrained nodes and typically one, or in some cases a couple of border routers [4]. The border router is also known the Gateway. If a node cannot communicate directly with the border router, it uses other nodes as intermediate nodes toward the border router. This process is handled by routing protocols in the network. So routing protocols play critical roles in delivering data from a node to the border router of IoT [5]. In this regard IETF has standardized a routing protocol for IoT, known as RPL [6]. RPL routing protocol enables users to define routing strategies according to their preferences about network requirements and metrics [7]. This facility is provided by Objective Function (OF) concept, which is also one of our focus points in this paper. OF defines how to decide about the suitability of a node in other to use it as a mean of achieving the network goal [8]. We have cited most important and up-to-date OFs in section III.

On the other hand, RPL and its OF were mainly designed for low-throughput networks with low dynamicity [9]–[11]. Hence, it encounters serious problems in a network under heavy load [10] or of high traffic dynamicity [9]. Considering this fact, in this paper, we focus on two important aspect of IoT networks, i.e. network lifetime and packet loss, in a network with heavy and highly dynamic load. We propose a new OF which recursively examines the lasts states of the chain of nodes in the path toward the root, while avoiding extra message passing requests in the network. We also put forward a new routing metric, which is devised for answering the dynamicity need in the network. besides, we present a new parent selection methodology. The paper structure is as follow: In section II, we give a brief explanation about RPL. Then in section III we cite most recent works in the scope of RPL protocol and also its objective function. Section IV is dedicated to the problem statement. in this section we explain the problem that we are going to solve. Then in section V, we present our contributions for solving the problems which are stated in Problem Statement part. Section VI is dedicated to explanation of the contribution. This section covers different parts of our contribution thoroughly. After that, in section VII, we evaluate our protocol in different scenarios, and compare the results with that of RPL. Finally, section VIII concludes the paper.

II. RPL: IPv6 ROUTING PROTOCOL FOR LOW-POWER AND LOSSY NETWORKS

In this section, we briefly introduce RPL routing protocol features and components. These definitions are extracted from IETF drafts.

A. RPL COMMUNICATION TYPES

There are three types of communications in RPL: multipoint to point (MP2P), which is from nodes toward gateway, point to multipoint (P2MP), which is from gateway to nodes, and point to point (P2P), which is from a node to another node in the network. In MP2P the traffic flows in an upward direction and is known as Inward Unicast Traffic. This type of communication is the most important and widely used type of communication in RPL [12], [13].In this paper, we focus on MP2P communication type.

The P2MP traffic, which is also called Outward Unicast Traffic, is provided by using destination advertisement mechanism [6], and is used in scopes like Industrial Automation [14]. In P2P type, the traffic either goes to the root then moves back toward the destination (non-storing mode), or it finds a way toward the destination from the first common parent with the destination node (storing mode).

B. RPL MESSAGE TYPES

There are three types of control message in RPL:

- DODAG Information Object (DIO): this message is issued by root and contains information about DAG instance, such as configuration parameters. This type is like the one that IPv6 uses for route advertisement [15].
- DODAG Information Solicitation (DIS): this message is issued to solicit DIO from a node and hence, is useful for probing neighbor nodes. The main application of this kind of message is to ask for DIO messages from neighboring nodes.
- Destination Advertisement Object (DAO): this message is used to send path information from nodes toward the root. It is issued in a unicast manner toward the selected parent (storing mode), or toward the DODAG root (non-storing mode). After reaching the root, a complete path will be established.

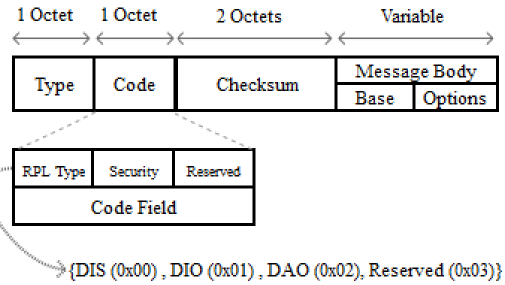

FIGURE 1. RPL control message format.

There is also another message type which is supported by RPL though not mentioned in message types, and that is Destination Advertisement Object Acknowledgment (DAO-Ack). This message is propagated by DAO receiver, as the acknowledgment of a DAO message, and contains information such as DAOSequence, RPLInstanceID, and Status. These types are defined under the category of ICMPv6 messages, and have the general structure of fig.1. These messages are mainly composed of a header part, which carries Type, Code and Checksum, and a body part, which contains the message base and options. The frequency of messages will increase in case of inconsistency, instability, or when there are new nodes that want to join the tree.

C. DODAG CONSTRUCTION

RPL uses DODAG concept, a Destination Oriented Directed Acyclic Graph, to construct routes in the network. At first, the gateway or the border router starts to propagate DIO messages through the network. The receiving node decides based on the Objective Function whether to choose the gateway as the parent or not. Each of nodes that chooses the gateway as the parent, starts to re-propagate the DIO message through the network. This process is repeated until all nodes in the network are covered by the constructed tree.

In a network, there may be multiple instances of DODAG simultaneously, with different instance IDs. This is because a node may decide to join multiple DODAGs due to its different requirements, i.e. lower delay, packet loss, or energy consumption. It is worth mentioning that when a new node wants to join the tree, it first asks for a DIO through a

DIS message. After receiving the DIO, it will reply by a DAO message, followed by a DAO-Ack by the parent in response to the DAO.

D. LOOP MANAGEMENT

RPL protocol tries to establish loop-free routes, but it does not guarantee the loop-freeness. In this regards, RPL follows some rules. The first rule is called max-depth rule. According to this rule, a node is not allowed to join a parent whose rank value is higher more than DAGMAXRankIncrease value. This rule prevents count-to-infinity problem but cannot fully avoid routing loops. Another rule, which is related to greedy parent selection, is that a node cannot choose a deeper parent in order to gain a lower rank. There are also mechanisms for loop detection which are based on Data Path Validation technique (a well-known technique in wireless communication scope). In data path validation technique, data packet transmissions and receptions are used as probes to detect loops in the network.

E. OBJECTIVE FUNCTION

Objective Function (OF) is one of the core concepts in RPL. OF defines how different metrics should be combined and translated into a rank, so that the protocol be able to use the rank to construct efficient routes. There are lots of routing metrics, like delay, packet loss, energy consumption, link quality, and so on. IETF has issued some drafts in this regard [16], [17]. Currently, there are two standard OFs for the RPL. The first one is OF0 [17]. OF0 works based on hop count metric. In this OF, rank is calculated by adding a value to the rank of the preferred parent. it does not consider link layer metrics, i.e. ETX, and its main goal is to bring connectivity to the network. The second standard OF is MRHOF [18]. MRHOF uses metrics like ETX[1] or latency as the basis of rank computation. Besides, it avoids instability caused by small metric changes.

F. MODES OF OPERATION

In RPL there are two main modes of operations: Storing mode and Non-storing mode.

- Storing mode: In this mode, when a DAO message is sent in a unicast manner toward a parent, the parent first stores the message information, then regenerates a DAO message which contains both previous DAO information and its own reachability information. After that, the new DAO message is forwarded to the parent. This process will be repeated until the message reaches DODAG root.
- Non-storing mode: In this mode, everything is like storing mode, with the only difference that instead of saving a DAO message, the parent only insert its reachability information into the DAO message and forwards it toward its parent. The rest of the process is the same as the storing mode.

[1]ETX (Expected Transmission Count) is the expected number of transmissions for a successful delivery of packets to the destination [19]

The decision about which mode to use is dependent on network requirements and also the available resources per node. For instance, if we want to have P2P communication in the network, we may experience a better performance by using storing mode.

III. RELATED WORKS

In RPL protocol, the topology construction and route selection is done on the basis of Objective Function (OF) and routing criteria. Objective function defines how to calculate the rank of a node and how to combine different criteria in process of rank computation. However, in RPL standard [6] there is no obligation to use a specific objective function or a set of criteria. So it is possible to manipulate the default OF and its parameters. This opportunity lets the designer of OF to be very flexible over choosing routing parameters; however, choosing the right parameters is not that easy. In this regard, IETF has made a set of recommendations for OF design [16]–[18]. RFC 6551 has put forward a set of routing criteria and constraints to be used in RPL but it has not described how to choose between them or how to combine them [6]. Hence, OF0 [17] or MRHOF [16] are commonly used as the default OF and in some cases they are merged with the proposed OF in [20].

Besides, some other implementations of RPL have used other parameters like Hop Count, ETX or a combination of these. For instance TinyRPL, which is a version of RPL implemented by TinyOS, has combined OF0 with Hop Counts. Contiki OS has also an implementation of RPL, known as ContikiRPL, which uses MRHOF as its default Objective Function, although it contains also OF0 implementation. As it can be seen, all different versions of OFs can be applied to an IoT network. So it is the OF designer who has the final say on which parameter to choose, or how to combine them in an OF.

Hence the proper definition of OF according to network needs and preferences is still a hot and open issue in the scope of RPL protocol. In this regard, Gaddour *et al.* [20] has presented a QoS-aware OF which works with fuzzy logic. In this method, fuzzy parameters are used to form a configurable routing protocol. The same approach have been followed in [21]–[23]. Another method is to combine Hop count with ETX, Remaining Power, and RSSI [24]. It is also possible to use MAC layer data to add another parameter for the packet loss [25]. However it's difficult to find the proper combination. Di Brachman [26] has investigated the effect of different sets of parameters and also different ways of combining them. In [27] Brachman has focused on multi-gateway networks and has presented an OF for the situation in which nodes should choose between different gateways.

Furthermore, a great deal of work is done to study RPL performance under different situations in different scenarios. For instance, Ko *et al.* [28] has inspected and compared RPL implementation of TinyOS with tree data collection protocol of TinyOS and has shown that they work almost the same with the only difference that TinyRPL supports IPv6. Additionally,

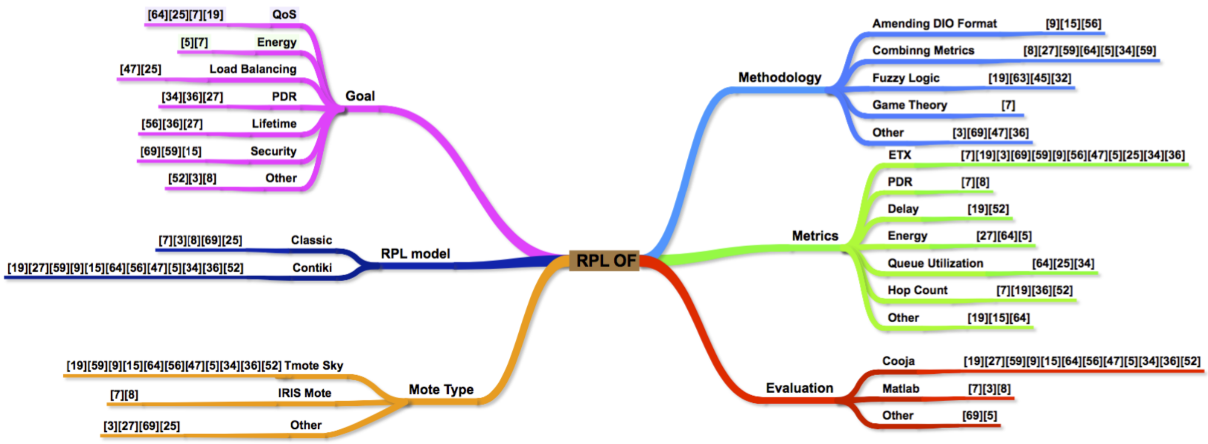

FIGURE 2. Research papers focusing on RPL objective function.

thorough comparisons have been done between TinyRPL and ContikiRPL [29] and also between RPL and LOAD protocols [30]. Kermajani and Gomez [31] has focused on improving RPL performance from the convergence point of view in process of tree formation in IEEE 802.15.4 based networks. RPL protocol has also been studied in COOJA simulator in disaster scenarios and its drawbacks have been discussed [32].

Afterwards, Clausen *et al.* [33] put forward some countermeasures to encounter the drawbacks, in such a way that it has solved reliability problem in bottom-up routes in RPL and also suggested a power adaptive method in which a sender transmits data directly to the receiver, using a high-power signal, instead of transmitting it to intermediate nodes.

There are also some works focusing merely on Objective Function. They have concentrated on different metrics like ETX [34]–[39], PDR[2] [40], [41], delay [42], [43], Energy [8], [44], [45], Queue Utilization [44], [46], [47], Hop Count [40], [42], [43], [48] and other metrics like trust [49]. We have summarized works that have focused on OF in fig 2. Beside these, there are works, focusing on Lifetime and/or packet loss, in a network that uses RPL as the routing protocol. In [50] Barbato *et al.* proposed a method which considers the energy status and also the position of the node in the network to decide about choosing or not choosing it as the forwarding node. It also defines different classes of traffic inside the network. But the performance of this method is not acceptable. In [51], a duty cycling method is proposed in which the author assumes that all nodes have equal amounts of resources and incoming traffic. But this is not always a valid assumption in IoT systems.

In [10], Kim *et al.* have presented QU-RPL, a queue utilization based extension of RPL. QU-RPL considers both queue utilization and hop distance to select parents. The authors have proved the superiority of their proposed protocol over RPL by evaluating it in real scenario. But in this protocol, the concentration of the authors is on balancing the load in high traffic and not on the dynamicity of the load, which is necessary in the scope of Intelligent IoT [52], IoT Service Provisioning [53], and Moving Object Tracking in IoT [54]. Besides, QU-RPL uses the same rank computation method as the standard RPL. The only difference is in rank propagation methodology, in which it wraps up the rank value, hop count, and queue utilization information together in DIO message. It also has no countermeasure for the state of Equality Illusion. We try to tackle with the problems of this work, which are also common problems between other related paper.

In [55], Kibria *et al.* present a multi-parent protocol which focuses on the bottlenecks in the network. This is done by introducing a new routing metric, which finds the bottlenecks in every path toward the root. But in rank computation, it simply uses ETX and parent rank, which may lead to a value that is not fully indicative of the condition of the node. In [56] Yang *et al.* has proposed a method which estimates the amount of traffic on a node and tries to make the network more stable by considering the traffic factor in routing. But it works only in a network with symmetric and bidirectional links, which is not always reasonable. In [57], Iova *et al.* puts forward a greedy approach which selects parents in a way that brings more stability to the network under high network dynamic. But the problem is that it requires frequent parent changes and consequently imposes lots of overhead on the network. In this paper, we are focusing on this problem and also DAG construction method under high traffic load and dynamicity.

IV. PROBLEM STATEMENT

RPL, chiefly designed for low power and lossy networks, has many outstanding features like loop-freeness, quick topology construction, self-healing mechanism, and low battery usage. But as it was initially designed for low traffic networks, it cannot deal with problems of a high traffic rate network. In other words, when the network traffic is heavy, RPL cannot handle it well, and the network faces with multiple problems

[2]Packet Delivery Ratio

such as high packet loss rate, energy depletion, and load imbalance. The problem becomes more devastating when a depleted node is the only intermediate node for a part of network toward the root. We categorize RPL problems under heavy and highly dynamic load as follow:

1- As the rank is computed by Objective Function (OF), which is OF0 and also MRHOF in standard RPL, a great deal of research is done to alter RPL objective functions, as cited in related works. But to the best of our knowledge, neither the standard RPL OFs nor other presented OFs take into account the previous parents of a node in succession. In other words, a node might seem to be eligible to become a parent, but the parent of the node, or another parent in the parent succession might be in the state of suffering from low remaining power or buffer space. This results in misappropriate parent selection in a heavy traffic network.

2- The route construction is done based on Rank. Once a node wants to join the network or change its parent, it chooses the one with smaller rank value. On the other hand, as we know, the rank of parents are computed in the initial stages of joining the network, while there may be lots of inconveniences in the path toward the route after rank computation. So in a network with high traffic dynamicity, rank value can not thoroughly reflect the last and real state of candidate parents. On the other hand, considering the dynamicity of the load is necessary for emerging concepts such as Intelligent IoT [52], [58], IoT Service Provisioning [53], [54], [59], and Moving Object Tracking in IoT [54], [60], [61]

3- Once a node joins a network, if the rank value that is advertised through DIO propagation is small, it abruptly attracts lots of other nodes, which are children of other parents at the time, toward itself. This phenomenon is called Thundering Herd Phenomenon [62]. To better understand the problem, consider a network like what is depicted in fig. 3. In this network, *a* and *b* are two parents holding some nodes in their sub-trees. Node *c* is another node that wants to join the network. The imaginary blue rectangles are showing the joint coverage areas. The left one is the joint coverage area between nodes *a* and *c*, and the right one is the joint coverage area between *b* and *c*. We suppose that the rank of nodes *a* and *b* are greater than that of node *c*. Then once the node *c* joins the network, and starts to broadcast DIO messages, it abruptly absorbs a significant portion of nodes in the joint coverage area with node *a*, and also with the one in common with node *b*.

This sudden change results in network instability, especially in large-scale networks with heavy traffic and high traffic dynamicity. Thundering Herd problem has been addressed in the new draft for optimization of parent node selection [62], but the countermeasure has never been proposed in any research papers or implemented in any version of RPL.

It is worth mentioning that in RPL standard [6], greedy parent selection has been addressed and avoided (Section 3.7.1 of the standard). Even though the thundering herd problem seems similar to the case of greedy parent selection, the latter occurs when a node has multiple parents and wants to exceptionally increase its rank (in spite of the nodes tendency to lower ranks), in order to increase the size of its parent set. However, thundering herd problems are related to the situation where a single-parent node wants to improve its status in the network by lowering its rank. Indeed, the case of multiple simultaneous parents and moving toward rank increase cannot occur in CLRPL.

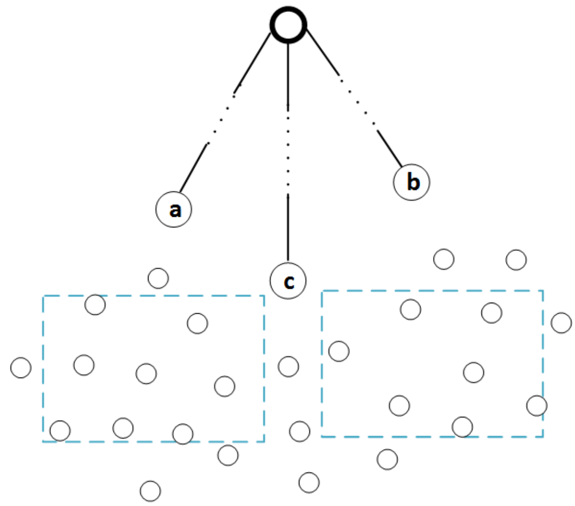

FIGURE 3. Thundering Herd phenomenon example.

V. CONTRIBUTIONS

We aim to put forward a modified version of RPL for IoT networks with high-speed sensor data streams. Managing the traffic in networks with high-speed sensor data streams is currently a hot topic in domains like smart cities and industrial networks [63], [64]. Our objective is to increases the network lifetime and decreases the packet loss of RPL in networks with high-speed streams. We do this by benefiting from DODAG structure and defining a new routing metric for RPL protocol. Our new protocol is able to balance the traffic and also energy level through the network. We evaluate our protocol through comprehensive simulations in Cooja simulator and comparing the result with standard RPL. Besides, we combat the Thundering Herd Phenomenon in the network. We can categorize our contributions into the followings:

1) We propose a new Objective Function, known as Context-Aware Objective Function (CAOF) which has the following attributes.
   a) It considers the remaining power of the parent chain in the route toward the root and combines it with ETX and parent rank.
   b) It avoids the thundering herd problem in the network.
2) We propose a new routing metric, known as Context-Aware Routing Metric (CARF). This metric has the following attributes:
   a) It considers Queue Utilization and also remaining power of parents' chain instead of a single parent.

b) It helps the network to have more power balanced and also load balanced routes.

3) We propose a parent selection mechanism with the following attributes:

a) It uses CARF as the decision criterion to choose the most suitable parent.

b) It avoids the Equality Illusion Problem, which will be explained later in section VI-C.

VI. CONTEXT-AWARE AND LOAD BALANCING RPL FOR HEAVY AND HIGHLY DYNAMIC NETWORK LOAD

This section is dedicated to different parts of our contribution. In section V we categorized our contribution into three parts. Here we will explain each part respectively.

A. CONTEXT-AWARE OBJECTIVE FUNCTION (CAOF)

Context is any information that can be used to typify the situation of an entity, in which an entity is an object, place or person, that is considered pertinent to the interaction between a user and an application, including the user and applications themselves [65]–[67]. Here, we have taken IoT nodes as entities. Moreover, a context-aware system is a system which provides pertinent information and/or services to the user, using context, while pertinency depends on user's task [65]–[67]. We want to make use of Context-Awareness concept to boost routing process of RPL. In resource constrained networks, nodes can provide small queue sizes. Consequently, a queue loss may happen a lot sooner than a link loss, under heavy load. So although ETX usually gives a reasonable estimation of the channel status, but using merely ETX, which is based on link loss, is not enough to understand the real status of the network. We need to combine it with another metric that is related to node queue. This way we can monitor both link status and queue status, and avoid a loss at early stages of a congestion.

There are also other metrics, used by different research papers, as cited in chapter III. But none of them takes into the consideration the last states of nodes in the path while calculating the rank value. Moreover, they calculate the rank value in a way that if there is any temporal suffering, i.e. power drainage, queue overflow, etc., for a node at the time of rank computation, it will affect the rank of all downward nodes permanently.

For this reason, we have presented a new objective function, known as CAOF, which recursively examines the lasts states of the chain of nodes in the path toward the root. Besides it lessens the effect of a node condition as the distance become greater, by giving a smaller weight to parent rank, in comparison to the newly calculated rank. This mechanism avoids any node to poison the rank chain in a path permanently. Moreover, CAOF avoids Thundering Herd Phenomenon by gradual shifting from a high rank value toward the real rank value. To the best of over knowledge, none of the previously presented objective functions can deal with these problems and this is the first research which presents an objective function that dynamically considers the network situation and also avoids the thundering herd effect in the network.

It is worth emphasizing that our purpose in this study is to increase network lifetime while decreasing the packet loss. To attain this goal we selected relevant metrics that influence network lifetime and packet loss. For the lifetime part we focused on the residual energy of the nodes, since as cited in [68], [69], [70],[71], and [72], residual energy of network nodes is a suitable metric when controlling network lifetime.

Also, [73], [74], [75], and [76] show that queue utilization is correlated with packet loss. So we considered queue utilization in the core of our proposed metric for decreasing packet loss. While ETX takes into account the link status between nodes, queue utilization and energy metrics are introduced to consider the node status. Hence, combining ETX, queue utilization and energy metrics provides a better understanding of the network situation thereby allows making wiser decisions towards achieving our goals.

To compute the rank of a node:

1) we define $\xi\,(n)$ as follow:

$$\xi(n) = \begin{cases} \dfrac{E_{init}(n) - E_{cur}(n)}{E_{init}(n)} & n = \text{root} \\ Max\left(\dfrac{E_{init}(n) - E_{cur}(n)}{E_{init}(n)}, \xi(n') \times \theta\right) & n \neq \text{root} \end{cases}$$
$$0 \leq \xi\,(n) \leq 1$$

In which n is the current node, n' is the parent node, $E_{init}(n)$is the initial power level of node and $E_{cur}(n)$ is the current power level of node n. $\frac{E_{init}(n)-E_{cur}(n)}{E_{init}(n)}$ is and index of Residual Energy of node n. $\xi\,(n)$ is the status of residual energy of chain of nodes in the path. In other words, it considers the remaining power of nodes in a recursive manner, while lessening the effect of parent remaining power as it gets farther down the path. θ, is the parameter used to lessen the power condition of a parent on the routing, so that the power depletion effect does not go farther than some levels in the network. According to simulations, the best result was achieved when θ is set to be 0.22 in CAOF. Greater value of $\xi\,(n)$ for a route is indicative of the route healthiness form remaining power point of view.

2) we combine ETX with parent rank and $\xi\,(n)$:

$$Rank(n') + (ETX(n, n') \times \xi(n))$$

here, ETX is defined as [77]:

$$ETX(n, n') = \frac{1}{d_{nn'} \times d_{n'n}}$$

in which $d_{nn'}$ is the measured probability that a data packet successfully travels from n to n', and $d_{n'n}$ is the measured probability that a data packet successfully travels back from n' to n. These probabilities will be calculated via probe messages.

3) We give weights to each part, so that we can give preference to new computed value rather than the old one:

$$\lambda \times Rank(n') + (1 - \lambda) \times (ETX(n, n') \times \xi(n))$$

The value of λ is set to be 0.8 in this paper.

4) Finally, we compute $Rank\ (n)$ as follow:

$$Rank(n) = \lambda \times Rank(n') + (1 - \lambda) \times (ETX(n, n') \times \xi(n))$$

In this formula, $\xi(n) \in [0, 1]$ is used as a weight factor for ETX to give more importance to nodes that are discharged less. This way, we differentiate between nodes with equal ETX values but different power conditions. In our rank computation method, the substantial factor is our proposed metric, $\xi(n)$, which differentiates our method from other rank computation methods. $\xi(n)$ brings the recursive information to the rank computation equation. In the case that the protocol does not consider the condition of previous parents, it may result in choosing a parent that has congestion problems in its way to the root, even if the selected parent is in a good status regarding the remaining power and the buffer space. We used parent information to shift the focus from merely a parent status to the chain-of-parents status thereby acquiring a comprehensive view in assessing the node eligibility to become a parent.

Besides, the prioritization factor λ gives a particular importance to new rank values so that the decisions are based on the recent status of the network.

Another point to consider is that in our method, a node with either very low residual energy or bad ETX, is interpreted as an inappropriate node. Imposing acceptable values of both metrics avoids a potential break in the path which can trigger a bottleneck. It is worth mentioning that the information about the queue utilization of a node, or its remaining power status, are included DIO messages before propagation. So we do not overload the network with an extra phase of message propagation or a new type of message.

After computing the rank, CAOF goes through the process of avoiding Thundering Herd Phenomenon. To achieve this goal, CAOF uses Algorithm 1.

Algorithm 1 CAOF Algorithm

Data: n, $\mathcal{DIO}_n$, $Rank(n)$
Result: Avoiding Thundering Herd Phenomenon
for $S \in 1, \ldots, |\mathcal{DIO}_n| - 1$ **do**
  for $R \in 1, \ldots, |\mathcal{DIO}_n| - S$ **do**
    if $Rank(\mathcal{DIO}_n(R)) < Rank(\mathcal{DIO}_n(R+1))$ **then**
      Swap $\mathcal{DIO}_n(R)$ **and** $\mathcal{DIO}_n(R+1)$
for $S \in 1, \ldots, |\mathcal{DIO}_n|$ **do**
  if $Rank(\mathcal{DIO}_n(S)) > Rank(n)$ **then**
    Set $Rank(n)$ to $Rank(\mathcal{DIO}_n(S))$ in DIO messages;
    wait()
  break

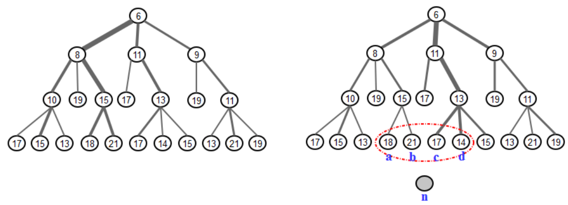

FIGURE 4. status of a part of network at time t(left), and t'(right).

This algorithm, first sorts the DIO_n, which is an array of all the DIO messages received by node n, that are in the same instance with the node n, and of the current RPL version. It is worth mentioning that as the array is not that big, it does not impose a high computation load on the node. After sorting, the algorithm starts to broadcast DIO messages, in a way that it first broadcasts a message with the maximum rank value among all received ranks. Then it waits an amount of time, equal to propagation frequency of standard RPL, and at the second phase of propagation, it sets the new rank value, equal to the second element of the DIO_n array. This way, the network will experience no rush toward a node that is advertising a low value rank, and consequently there will be no thundering herd problem in the network. After lowering the rank value step by step, it will be set to its real value, which was computed initially by the node.

B. CONTEXT-AWARE ROUTING METRIC (CARF)

In process of choosing a parent, RPL uses Rank as the decision criterion. Smaller value of a node rank will be interpreted as more suitability of that node for becoming a parent. But making a decision merely on the basis of the rank will not lead the node to the best possible choice, especially when the network is under the heavy traffic. In RPL, the rank of a node is computed as the node joins the network, and it changes only if the node changes its parent. So until the next probable parent change, the node has the same rank value as it had at the beginning of joining the parent. But the problem is that, this value cannot be a good criterion to judge about a network with varying situation of the remaining buffers due to the high and dynamic traffic.

To better understand the problem, we have illustrated a sample network snapshot in fig. 4. This figure shows a part of a network at two different times. The thickness of the lines is indicative of level of traffic intensity, and numbers on the node are rank value of that node. OF of this network considers queue utilization of parent as one of the factors for calculating the rank. The tree on the left shows calculated rank values after the DODAG construction, and at time t. After a while, at time t', node n wants to join the tree as a new node. At the right tree, the node n and 4 other nodes in its communication range are shown and named a, b, c, and d. Naturally, node n selects node d as its parent due to the lower value of rank that d holds. But logically this is not the best choice; Because as it can be seen, node d, contrary to its good status at time t, is not at a suitable parent-chain load status at t'. So current OF, in spite of considering queue utilization at rank computation, could not lead the node n to the best parent choice. In RPL, OF and also parent selection mechanism are even worse, because RPL is neutral to queue utilization and load dynamicity. To solve this problem, we have presented a Context-Aware Routing Metric (CARF), which consider the remaining buffers of chain of nodes in the route at the time of decision. to do this:

1) we define $\omega(n)$ as follow:

$$\omega(n) = \begin{cases} n = root & Q(n) \\ n \neq root & Max\left(Q(n), \omega(n') \times \theta\right) \end{cases}$$
$$0 \leq \omega(n),\ Q(n) \leq 1$$

$\omega(n)$ shows the status of queue chain in the path. It computes the occupancy of nodes queues in a recursive manner, while lessening the effect of queue occupancy as it gets farther down the path. θ is the same parameter as used before. $Q(n)$ is queue utilization indicator and is defined as

$$Q(n) = \frac{space\ occupied\ in\ queue\ of\ node\ (n)}{total\ queue\ size\ of\ node\ (n)}$$

2) We compute $\eta(n)$ as follow:

$$\eta(n) = \frac{1}{r}\sum_{i=1}^{r}\left(\omega_i(n') - \varpi\right)^2$$

where

$$\varpi = \frac{\sum_{i=1}^{r}\omega_i(n')}{r}$$

$\eta(n)$ is Network Traffic Dynamicity Index, and is indicative of how frequently did the traffic change during past r values for $\omega(n)$. The value of r is defined based on memory constraints in the network. In this paper, we set r to 5, considering constraints of Tmote sky nodes used in our network.

3) we compute ς as follow:

$$\varsigma(n) = \eta(n) \times Rank(n) + (1 - \eta(n)) \times \omega(n)$$

$\varsigma(n)$ is the metric which will be used to decide about which parent to choose. This metric can lead to a better decision in a network with heavy traffic. We make use of this metric in our parent selection mechanism, which will be explained afterward.

C. PARENT SELECTION MECHANISM

To increase the network lifetime while decreasing the packet loss, we consider the situation in which a node wants to change its parent due to some inconveniences, i.e. power drainage and/or buffer overflow, in parent node or the state in which a node wants to join the DODAG for the first time. In this situation, it should select between the candidate set of parents, and choose the most suitable one. In RPL, this selection is done on the basis of rank of the parents. But as we explained earlier, this metric is not suitable enough to choose a parent.

Moreover, none of the research papers in the scope of RPL routing protocol has considered the number of children of a node, as a metric in rank computation. The reason behind that is that generally, a rank is computed when a new node wants to join a network. In that case, the new node does not still have any children. So it is not possible to consider the number of children of the node as a metric for rank computation.

Consider a network with multiple nodes, each one with a different number of children. Basing the parent selection decision on only the rank value is not accurate. Even if the ranks of the nodes are equal, they may have a different number of children, and hence different suitability level for being a parent. We refer to this state as *Equality Illusion* state. In other words, *Equality Illusion* is the state of having multiple candidate parents with the same rank value but different numbers of children.

In this regard, we put forward Algorithm 2 as a new parent selection mechanism, which makes use of CARF and also some other pieces of information about parents, to choose between them, while avoiding Equality Illusion Problem. It also avoids routing loops by a simple, yet effective solution.

Algorithm 2 Parent Selection

Data: $\mathcal{P}_n$ ▷ `Set of Candidate Parents for node n`
Result: P^{sel} ▷ `Selected Parent for node n`
$P^{sel} \leftarrow \mathcal{P}_n[1]$
for $P \in 1, \ldots, |\mathcal{P}_n|$ **do**
  if $h(P) \leq h(n)$ ***or*** $DODAG(P) \neq DODAG(n)$ **then**
    if $\varsigma(P) < \varsigma(P^{sel})$ **then**
      $P^{sel} \leftarrow P$
  if $\varsigma(P) == \varsigma(P^{sel})$ ***and*** $NoC(P) < NoC(P^{sel})$ **then**
    $P^{sel} \leftarrow P$
Return P^{sel}

In this algorithm, by NoC we mean Number of Children. This algorithm first sets the first element of parent array as the initial value for the selected parent. Then for each node in the candidate parent set it checks the hop count value of that node to make sure it is not greater than hop count of the node that is searching for the new parent. This simple though effective procedure avoids a node in the DODAG to select one of its successors as the parent, and consequently, prevents the node from creating a loop.

After that, the algorithm checks ς value of the candidate parent, as the first decision criterion (second *if statement*). In case that the ς of the candidate parent is smaller than ς value of currently selected parent, the algorithm sets candidate parent as the selected parent. As explained earlier, ς considers Rank and ω values. On the other hand, the rank value is calculated using ETX and ξ. In a nutshell, the first decision on parent selection considers ETX, residual energy, and queue utilization metrics.

If no decision is possible using these three metrics, then we are a state an equality illusion (explained in VI-C). Thus,

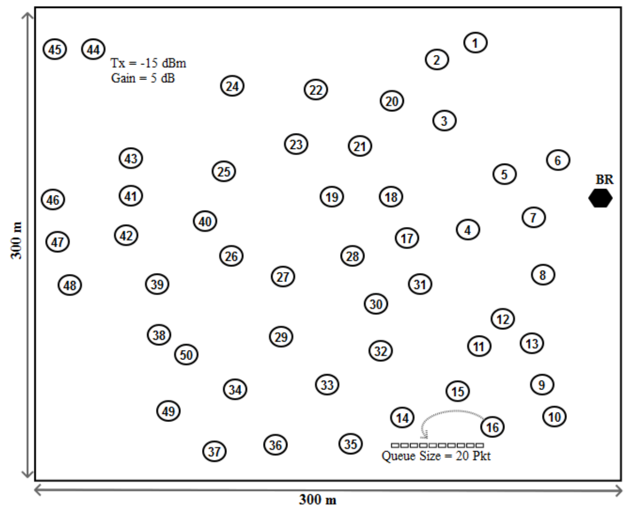

FIGURE 5. Simulation scenario.

the second decision criterion is NoC (third *if statement*). The one with the smallest number of children will be selected as the parent. Finally, P^{sel} is returned as the selected parent, after checking all nodes in the candidate parent set.

VII. EVALUATION

To evaluate our protocol, we simulated CLRPL with Cooja Simulator, a widely used simulator in the field of IoT developed by Contiki [78]. We compare CLRPL with ContikiRPL. We set MRHOF as the OF of contikiRPL, and the parameters of the protocol are tuned according to CLRPL ones (Table 1). In our scenario, we have 50 nodes with one border router, deployed in a 300m × 300m area. The border router acts as the root. Nodes type is Tmote Sky with MSP430 microcontroller, with 2.4GHz IEEE 802.15.4 Chipcon Wireless Transceiver, and they follow the energy consumption model of the CC2420 transceiver. The motes have 10k of RAM and 48k of Flash memory. They use Contiki OS 2.7 as the operating system and IEEE 802.15.4 communication protocol to communicate with each other. We used UDGM[3] of Cooja with distance loss, for the Radio Medium.

We have disabled duty cycling in this scenario, to reach heavy load in the network and we have used FIFO queue with a capacity of 20 packets. Fig. 5 depicts the simulation scenario and Table 1 delineates the simulation parameters. Besides, a varied number of nodes and traffic rates are considered to evaluate our protocol under different situations. For example in fig. 15 we have increased the number of nodes up to 100. We used 15, 20,30,40,80,100 ppm (packet per minute) in different scenarios. In fig. 9, fig. 10, fig. 11, fig. 12, fig. 15, fig. 16 and fig. 17, the traffic rate is 30 ppm, which is the same traffic rate that is used for heavy traffic networks in other papers like [9] and [79]. In fig. 18, we plotted the network condition for both 15 ppm and 30 ppm traffic rates. In fig. 6, fig. 13, and fig. 14, the traffic rate is changing from 20 ppm

[3]Unit Disk Graph Medium

TABLE 1. Simulation parameters.

Parameter	Value
Number of Nodes	50
Simulation area	300m x 300m
Traffic Type	CBR
Traffic Rate	15,20,30,40,80,100 Pkt/min
Energy Consumption	CC2420 transceiver datasheet
Packet format	IPv6
Mote Type	Tmote Sky
Chanel check rater	16,32 per second
Initial Energy	5 J
Mac Protocol	CSMA
Transmission Power	-15 dBm
Gain	5dB
Queue Size	20 Pkts
Transceiver	CC2420
Microcontroller	MSP430
Queue Type	FIFO
Rx Current Consumption	23 mA
Tx Current Comsumtion	21 mA
Microprocessor Current at Vcc=3V	0.6 mA
Distance Loss	90% Rx Ratio
Transmission Range	100m
ETX_SCALE	100
ETX_ALPHA	90
LOLLIPOP Max	255
LOLLIPOP Init	240

to 100 ppm to evaluate the protocol form low traffic rates to extreme cases.

First comparison is done from the view point of queue loss ratio. Fig. 6 shows two protocols, under different traffic loads. In this scenario traffic load changes from 1000 ppm (packet per minute) to 5000 ppm. The outcome reveals that CLRPL lessens the queue loss ratio up to 18% in the network. But another important aspect of fig. 6 is the worst case of queue loss ratio between nodes in different traffic loads. As it can be seen, there are nodes in the scenario that have up to around 73% of queue loss, using RPL, while this amount can be reduced to 40% of queue loss, using CLRPL. This can be interpreted as the ability of CLRPL in creating a more balanced DODAG from network load point of view.

But to scrutinize it more, we go over a sample snapshot of the network during simulation to know more about the condition of nodes in the network, and the way each protocol manage network transmissions. Fig. 7 shows the network topology in case of using RPL. Number of each node is

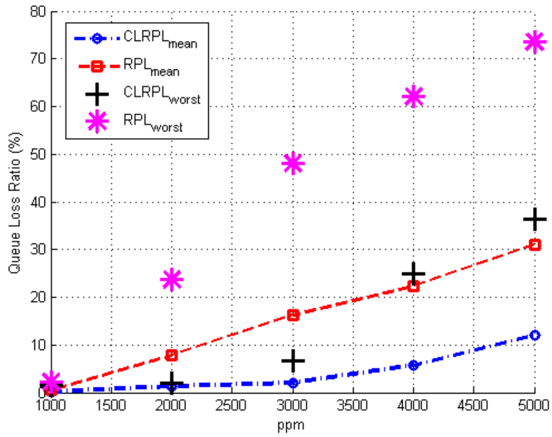

FIGURE 6. Queue loss ratio under different loads.

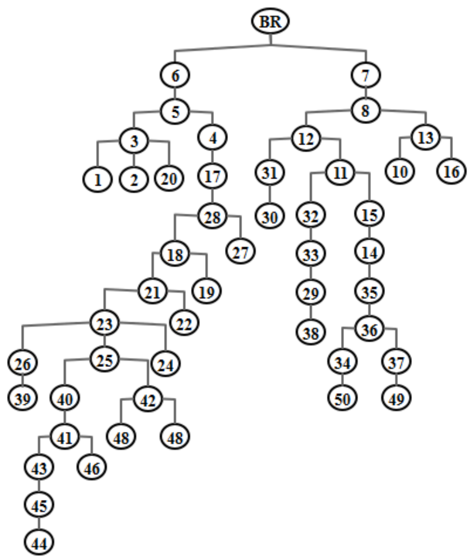

FIGURE 7. Network topology made by RPL.

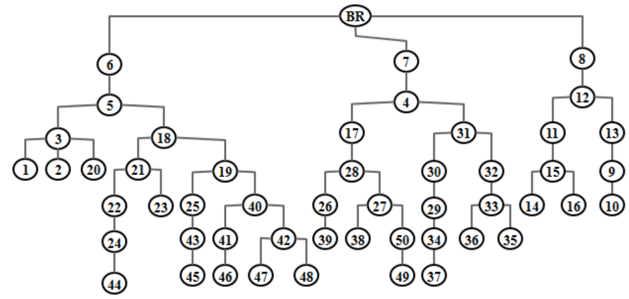

FIGURE 8. Network topology made by CLRPL.

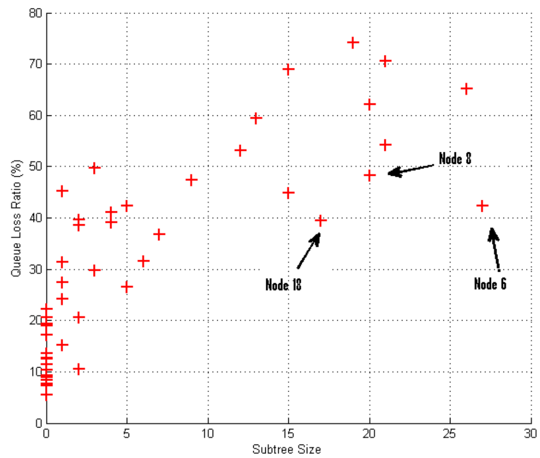

FIGURE 9. Subtree size to Queue loss ratio in RPL.

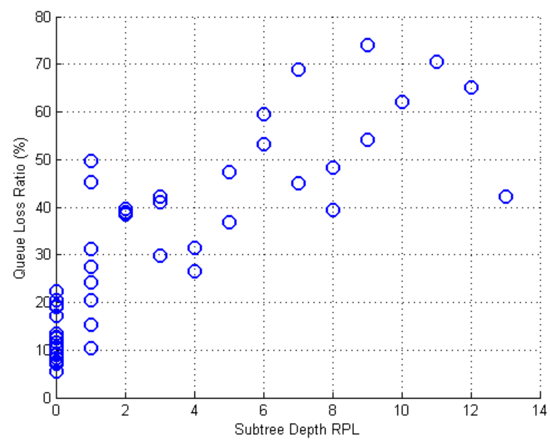

FIGURE 10. Subtree depth in RPL to Queue loss ratio.

written on it. In this figure, some nodes like node 28 or 12, have to handle lots of nodes in process of packet forwarding, while others do not experience such a heavy load during network lifespan. So the imbalancy of the RPL tree brings congestion and packet loss to the network which affects the network lifetime as well. But as fig. 8 shows, those nodes which were previously under severe load, seem to be in a better state in CLRPL as the consequence of a better DODAG construction.

To scrutinize more about the reason of queue loss, we have plotted the network from both subtree size and subtree depth points of view. Here, by depth, we mean the maximum number of hops between a parent and nodes of its subtree. Fig. 9 and fig. 10 depicts queue loss ratio in comparison with subtree size and subtree depth respectively.

In fig. 9 as you move toward bigger subtree size, you see that there is higher ratio of packet loss, so we can infer that there is a correlation between size of subtree and queue loss ratio. But this correlation is not tight, since there are also lots of nodes with bigger subtree sizes, e.g. node 6, node 8, node 18, that experience low amount of queue loss. Fig. 10 also depicts the same level of correlation between the subtree size and queue loss ratio. We plotted the network from two other perspectives in fig. 11 and fig. 12. In fig. 11 we investigate the potential relationship between Hop Count and Queue Loss Ratio. As it can be seen, different ratios of packet loss are scattered through different numbers of hop count. So we can infer there is no tight relation between Hop Count and Packet Loss Ratio.

After these evaluations, we tried to look at the problem from another perspective. In the new perspective we

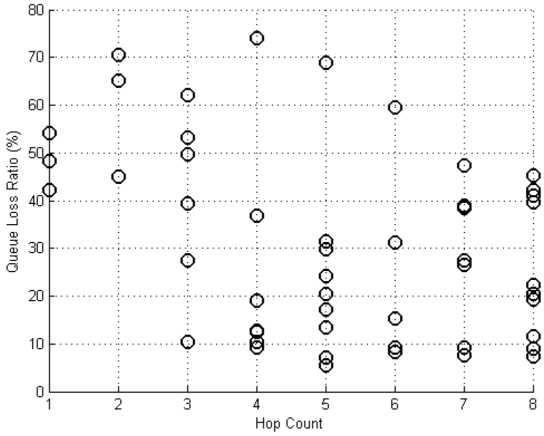

FIGURE 11. Distance from border router to Queue loss ratio in RPL.

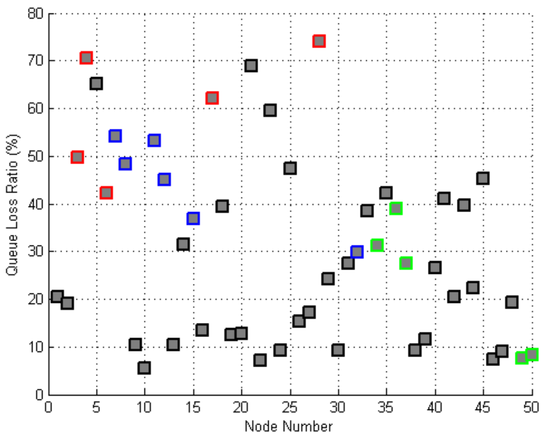

FIGURE 12. Nodes Queue loss ratio in the topology made by RPL.

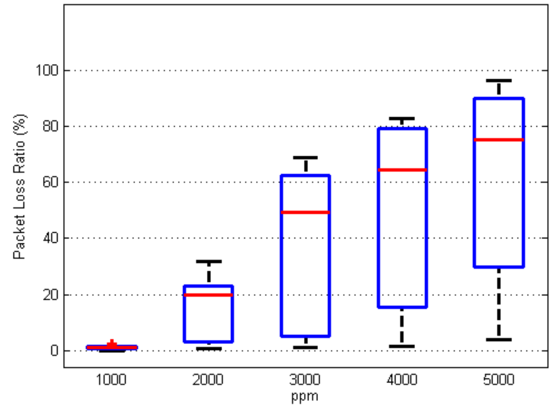

FIGURE 13. Packet loss ratio statistics of RPL at different traffic loads.

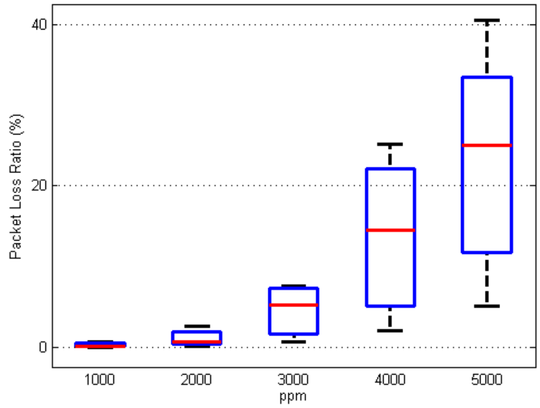

FIGURE 14. Packet loss ratio statistics of CLRPL at different traffic loads.

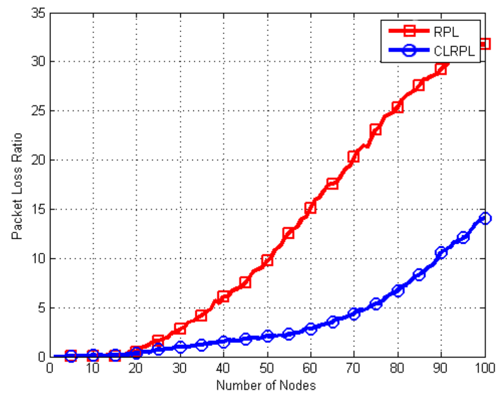

FIGURE 15. Packet loss ratio at different network sizes.

investigated the effect of parent status on its children queue loss. Fig. 12 show each node of the fig. 7 according to its queue loss ratio. This figure reveals that not only subtree size of a node can affect its packet delivery ratio, but also the situation of the parent under the load can affect a node queue loss ratio a lot. In fig. 12 we have indicated some sets of the parent-child chain in different colors. Nodes in red are *parent-child set 1*, nodes in blue are *parent-child set 2*, and nodes in green are *parent-child set 3*. We can see that each set of the parent-child seems to be approximately at the same condition.

For instance, nodes in *parent-child set 1*, are in a different queue loss condition than leaf nodes (nodes with no child) and also holding a worse queue loss ratio than nodes in *parent-child set 2*, and also than nodes in *parent-child set 3*. This is where CLRPL outperforms RPL in deciding about how to make the tree. We have plotted RPL and CLRPL under different traffic loads in fig. 13 and fig. 14. The result reveals that CLRPL decreases packet loss ratio significantly for most of the nodes. Considering the limited embedded queue in IoT nodes, most of the packet loss is the result of inappropriate parent selection when the network uses RPL as the routing protocol, while CLRPL tries to handle this problem with proper measures.

In another scenario we evaluated our protocol at different network sizes. We have evaluated it for up to 100 nodes and under a traffic of 3000 ppm. Fig. 15 shows the result of this evaluation. As the figure shows, the packet loss problem even becomes more serious when the number of nodes grows. In this case, RPL experiences up to 32% of packet loss, while that of CLRPL is under 15%.

On the other hand, misappropriate parent selection can also affect network lifetime a great deal. This is because of the

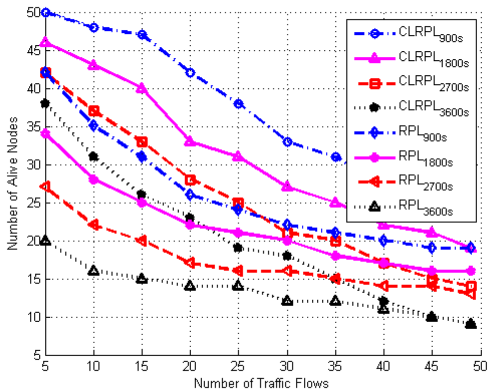

FIGURE 16. Number of alive nodes at different traffic flows.

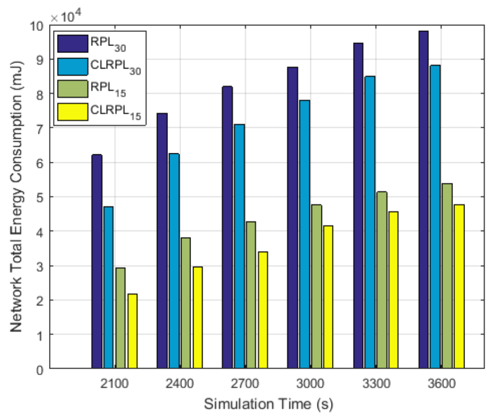

FIGURE 17. Network energy consummation.

fact that the nodes which are excessively under heavy traffic will face the problem of power depletion earlier than the other nodes. So at the time of a node depletion, even if there be an acceptable level of energy at the rest of the network, it would be interpreted as dead. This can even cause more serious problems when the traffic is unbalanced. So to evaluate the protocol from the lifetime perspective, we compare it with RPL under different traffic flows and varying the network load. A traffic flow is a stream of packets for 100 seconds, at a data rate of either 15 or 30 packets per minute, depending on the scenario. Fig. 16 compares the number of alive nodes in cases of using RPL and CLRPL under different traffic flows at several simulation times. Generally, a node with less than 5% of initial energy is considered to be non-functional and dead [8]. Otherwise, it is interpreted as alive.

As the fig. 16 shows, CLRPL performs on average 10% better than RPL in maintaining the nodes alive in the network. Moreover, fig. 17 and fig. 18, are showing the network total energy consumption and network lifetime respectively. In fig. 17, the indexes show the number of traffic flows. For example, RPL_3_0 depicts the RPL results when there is 30 simultaneous traffic flows in the network. The result reveals the superiority of CLRPL over RPL in decreasing the energy consumption. However, RPL seems to have a

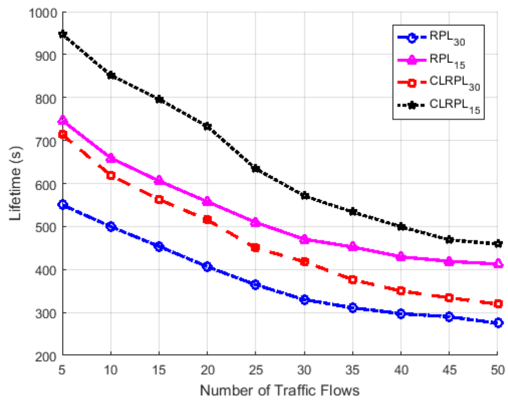

FIGURE 18. Network lifetime.

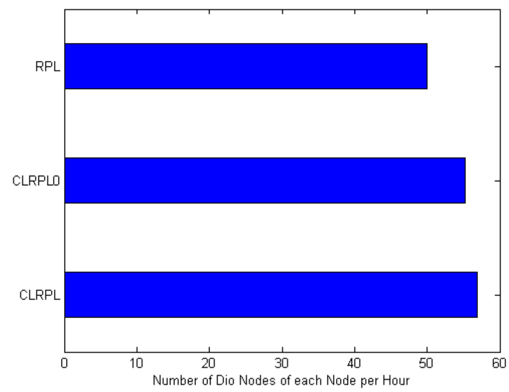

FIGURE 19. DIO overhead.

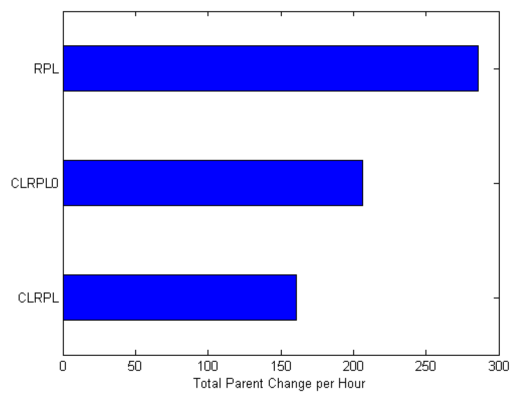

FIGURE 20. Parent change per Hour.

slower consumption growth as the simulation time increases. In fig. 18, the indexes show the number of packets per minute. Again, CLRPL has resulted in longer network lifetime, although RPL reveals a less steep slope in the results.

Simultaneously, we have also evaluated CLRPL from the overhead perspective as shown in Fig. 19 and fig. 20 are dedicated to the results. In these figures, by CLRPL0 we mean the situation in which we omit thundering herd management mechanism from CLRPL, while the rest remain the same. Fig. 19 reveals that CLRPL does not impose too much

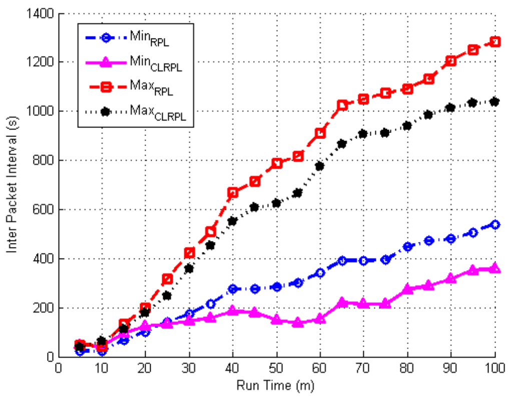

FIGURE 21. Inter-packet intervals.

of overhead on the network in comparison with RPL. Also in fig. 20, we observe that CLRPL can significantly lower the number of parent change in the network. Finally, we compare inter-packet intervals of two protocols. A lower inter-packet interval in a protocol can help achieving a lower end to end delay. Fig. 21 compares the inter-packet intervals of RPL and CLRPL. the result reveals that RPL has better Min values, but worse Max ones. This result leads us to the conclusion that in addition to a lower Max inter-packet interval value in CLRPL, this protocol also experiences a lower inter-packet interval difference among different nodes in comparison with RPL.

VIII. CONCLUSION

In this paper, we dealt with the problems of RPL routing protocol under heavy and dynamic load with the focus on packet loss and network lifetime. We identified that standard RPL cannot efficiently handle the heavy and dynamic loads. To solve the problem, we proposed a context-aware and load balancing protocol, called CLRPL, which considers the status of a parent-chain before selecting the last parent of the chain as the selected parent for a node. This way, we tried to balance the load in the network. We considered remaining queue and also energy level of candidate parents beside ETX metric. We also prevented the problem of rushing toward a suitable parent, which faces the network with instability and high control message rate problem. We evaluated our protocol in Cooja in different scenarios, and we proved that CLRPL outperforms RPL significantly, while not imposing too much of overload on the network.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey, " *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] H.-S. Kim, J.-S. Bang, and Y.-H. Lee, "Distributed network configuration in large-scale low power wireless networks," *Comput. Netw.*, vol. 70, pp. 288–301, Sep. 2014.

[3] N. Kushalnagar, G. Montenegro, and C. Schumacher, *IPv6 Over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals*, document RFC 4919, Aug. 2007.

[4] H.-S. Kim, H. Im, M.-S. Lee, J. Paek, and S. Bahk, "A measurement study of TCP over RPL in low-power and lossy networks," *J. Commun. Netw.*, vol. 17, no. 6, pp. 647–655, 2015.

[5] Z. Yang, S. Ping, H. Sun, and A.-H. Aghvami, "CRB-RPL: A receiver-based routing protocol for communications in cognitive radio enabled smart grid," *IEEE Trans. Veh. Technol.*, vol. 66, no. 7, pp. 5985–5994, Jul. 2017.

[6] T. Winter *et al.*, *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*, document RFC 6550, Mar. 2012.

[7] A. Mayzaud, R. Badonnel, and I. Chrisment, "A distributed monitoring strategy for detecting version number attacks in RPL-based networks," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 2, pp. 472–486, Jun. 2017.

[8] A. Hassan, S. Alshomrani, A. Altalhi, and S. Ahsan, "Improved routing metrics for energy constrained interconnected devices in low-power and lossy networks," *J. Commun. Netw.*, vol. 18, no. 3, pp. 327–332, Jun. 2016.

[9] Y. Tahir, S. Yang, and J. McCann, "BRPL: Backpressure RPL for high-throughput and mobile IoTs," *IEEE Trans. Mobile Comput.*, vol. 17, no. 1, pp. 29–43, Jan. 2018.

[10] H.-S. Kim, H. Kim, J. Paek, and S. Bahk, "Load balancing under heavy traffic in RPL routing protocol for low power and lossy networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 964–979, Apr. 2017.

[11] Z. Sheng, S. Yang, Y. Yu, A. Vasilakos, J. McCann, and K. Leung, "A survey on the IETF protocol suite for the Internet of Things: Standards, challenges, and opportunities," *IEEE Wireless Commun.*, vol. 20, no. 6, pp. 91–98, Dec. 2013.

[12] J. Martocci, P. D. Mil, N. Riou, and W. Vermeylen, *Building Automation Routing Requirements in Low-Power and Lossy Networks*, document RFC 5867, Jun. 2010.

[13] M. Dohler, T. Watteyne, T. Winter, and D. Barthel, *Routing Requirements for Urban Low-Power and Lossy Networks*, document RFC 5548, May 2009.

[14] A. Brandt, J. Buron, and G. Porcu, *Home Automation Routing Requirements in Low-Power and Lossy Networks*, document RFC 5826, Apr. 2010.

[15] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, *Neighbor Discovery for IP Version 6 (IPv6)*, document RFC 4861, Sep. 2007.

[16] O. Gnawali and P. Levis, *The ETX Objective Function for RPL*, document draft-gnawali-roll-etxof-00, Working Draft, IETF Secretariat, Internet-Draft, Feb. 2010.

[17] P. Thubert, *Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)*, document RFC 6552, Mar. 2012.

[18] O. Gnawali and P. Levis, *The Minimum Rank With Hysteresis Objective Function*, document RFC 6719, Sep. 2012.

[19] Y. Jin, H. Miao, Q. Ge, and C. Zhou, "Expected transmission energy route metric for wireless mesh senor networks," *Int. J. Digit. Multimedia Broadcast.*, vol. 2011, pp. 1–7, Apr. 2011.

[20] O. Gaddour, A. Koubâa, N. Baccour, and M. Abid, "OF-FL: QoS-aware fuzzy logic objective function for the RPL routing protocol," in *Proc. 12th Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, 2014, pp. 365–372.

[21] I. H. Urama, H. Fotouhi, and M. M. Abdellatif, "Optimizing RPL objective function for mobile low-power wireless networks," in *Proc. IEEE 41st Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, vol. 2. Jul. 2017, pp. 678–683.

[22] H. Lamaazi and N. Benamar, "RPL enhancement using a new objective function based on combined metrics," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, 2017, pp. 1459–1464.

[23] P. O. Kamgueu, E. Nataf, and T. N. Djotio, "On design and deployment of fuzzy-based metric for routing in low-power and lossy networks," in *Proc. IEEE 40th Local Comput. Netw. Conf. Workshops (LCN Workshops)*, Oct. 2015, pp. 789–795.

[24] P. Karkazis *et al.*, "Design of primary and composite routing metrics for RPL-compliant wireless sensor networks," in *Proc. Int. Conf. Telecommun. Multimedia (TEMU)*, 2012, pp. 13–18.

[25] P. Di Marco, C. Fischione, G. Athanasiou, and P.-V. Mekikis, "MAC-aware routing metrics for low power and lossy networks," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 13–14.

[26] A. Brachman, "RPL objective function impact on LLNs topology and performance," in *Proc. NEW2AN*, 2013, pp. 340–351.

[27] M. O. Farooq, C. J. Sreenan, K. N. Brown, and T. Kunz, "Design and analysis of RPL objective functions for multi-gateway ad-hoc low-power and lossy networks, " *Ad Hoc Netw.*, vol. 65, pp. 78–90, Oct. 2017.

[28] J. Ko, S. Dawson-Haggerty, O. Gnawali, D. Culler, and A. Terzis, "Evaluating the performance of RPL and 6LoWPAN in TinyOS," in *Proc. Workshop Extending Internet Low Power Lossy Netw. (IP+SN)*, vol. 80. 2011, pp. 85–90.

[29] J. Ko *et al.*, "Industry: Beyond interoperability—Pushing the performance of sensor network IP stacks," in *Proc. 9th ACM Conf. Embedded Netw. Sensor Syst. (SenSys)*, 2011, pp. 1–11.
[30] U. Herberg and T. Clausen, "A comparative performance study of the routing protocols LOAD and RPL with bi-directional traffic in low-power and lossy networks (LLN)," in *Proc. 8th ACM Symp. Perform. Eval. Wireless Ad Hoc, Sensor, Ubiquitous Netw. (PE-WASUN)*, 2011, pp. 73–80.
[31] H. Kermajani and C. Gomez, "On the network convergence process in RPL over IEEE 802.15.4 multihop networks: Improvement and trade-offs," *Sensors*, vol. 14, no. 7, pp. 11993–12022, 2014.
[32] N. Accettura, L. A. Grieco, G. Boggia, and P. Camarda, "Performance analysis of the RPL routing protocol," in *Proc. IEEE Int. Conf. Mechatron.*, Apr. 2011, pp. 767–772.
[33] T. Clausen, U. Herberg, and M. Philipp, "A critical evaluation of the IPv6 routing protocol for low power and lossy networks (RPL)," in *Proc. IEEE 7th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2011, pp. 365–372.
[34] A. Aijaz, H. Su, and A.-H. Aghvami, "CORPL: A routing protocol for cognitive radio enabled AMI networks," *IEEE Trans. Smart Grid*, vol. 6, no. 1, pp. 477–485, Jan. 2015.
[35] L. Ye, V. Fodor, T. Giannetsos, and P. Papadimitratos, "Path metric authentication for low-power and lossy networks," in *Proc. 1st ACM Int. Workshop Cyber-Phys. Syst. Smart Water Netw. (CySWater)*, 2015, Art. no. 5.
[36] D. Shreenivas, S. Raza, and T. Voigt, "Intrusion detection in the RPL-connected 6LoWPAN networks," in *Proc. 3rd ACM Int. Workshop IoT Privacy, Trust, Secur. (IoTPTS)*, 2017, pp. 31–38.
[37] F. Boubekeur, L. Blin, R. Leone, and P. Medagliani, "Bounding degrees on RPL," in *Proc. 11th ACM Symp. QoS Secur. Wireless Mobile Netw. (Q2SWinet)*, 2015, pp. 123–130.
[38] M. Qasem, A. Al-Dubai, I. Romdhani, B. Ghaleb, and W. Gharibi, "A new efficient objective function for routing in Internet of Things paradigm," in *Proc. IEEE Conf. Standards Commun. Netw. (CSCN)*, Oct./Nov. 2016, pp. 1–6.
[39] M. Mamdouh, K. Elsayed, and A. Khattab, "RPL load balancing via minimum degree spanning tree," in *Proc. IEEE 12th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2016, pp. 1–8.
[40] M. Barcelo, A. Correa, J. L. Vicario, and A. Morell, "Cooperative interaction among multiple RPL instances in wireless sensor networks, " *Comput. Commun.*, vol. 81, pp. 61–71, May 2016.
[41] M. Barcelo, A. Correa, J. L. Vicario, A. Morell, and X. Vilajosana, "Addressing mobility in RPL with position assisted metrics," *IEEE Sensors J.*, vol. 16, no. 7, pp. 2151–2161, Apr. 2016.
[42] O. Gaddour, A. Koubâa, and M. Abid, "Quality-of-service aware routing for static and mobile IPv6-based low-power and lossy sensor networks using RPL," *Ad Hoc Netw.*, vol. 33, pp. 233–256, Oct. 2015.
[43] M. R. Parsaei and A. R. Parnian, "IPv6 based routing in building automation network," in *Proc. 2nd Int. Conf. Knowl.-Based Eng. Innov. (KBEI)*, 2015, pp. 1025–1031.
[44] Z. Wang, L. Zhang, Z. Zheng, and J. Wang, "An optimized RPL protocol for wireless sensor networks," in *Proc. IEEE 22nd Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Dec. 2016, pp. 294–299.
[45] M. Banh, N. Nguyen, K.-H. Phung, L. Nguyen, N. H. Thanh, and K. Steenhaut, "Energy balancing RPL-based routing for Internet of Things," in *Proc. IEEE 6th Int. Conf. Commun. Electron. (ICCE)*, Jul. 2016, pp. 125–130.
[46] J. Guo, P. Orlik, and K. Ishibashi, "Resource aware hierarchical routing in heterogeneous wireless IoT networks," in *Proc. 8th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, 2016, pp. 599–604.
[47] H. A. A. Al-Kashoash, Y. Al-Nidawi, and A. H. Kemp, "Congestion-aware RPL for 6LoWPAN networks," in *Proc. Wireless Telecommun. Symp. (WTS)*, 2016, pp. 1–6.
[48] M. M. Khan, M. A. Lodhi, A. Rehman, and F. B. Hussain, "A multi-sink coordination framework for low power and lossy networks," in *Proc. Int. Conf. Ind. Inform. Comput. Syst. (CIICS)*, 2016, pp. 1–5.
[49] N. Djedjig, D. Tandjaoui, F. Medjek, and I. Romdhani, "New trust metric for the RPL routing protocol," in *Proc. 8th Int. Conf. Inf. Commun. Syst. (ICICS)*, 2017, pp. 328–335.
[50] A. Barbato, M. Barrano, A. Capone, and N. Figiani, "Resource oriented and energy efficient routing protocol for IPv6 wireless sensor networks," in *Proc. IEEE Online Conf. Green Commun. (OnlineGreenComm)*, Oct. 2013, pp. 163–168.
[51] O. Landsiedel, E. Ghadimi, S. Duquennoy, and M. Johansson, "Low power, low delay: Opportunistic routing meets duty cycling," in *Proc. ACM/IEEE 11th Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, Apr. 2012, pp. 185–196.
[52] Y. Sahni, J. Cao, S. Zhang, and L. Yang, "Edge mesh: A new paradigm to enable distributed intelligence in Internet of Things," *IEEE Access*, vol. 5, pp. 16441–16458, 2017.
[53] R. Y. Zhong, L. Wang, and X. Xu, "An IoT-enabled real-time machine status monitoring approach for cloud manufacturing, " *Procedia CIRP*, vol. 63, pp. 709–714, Jul. 2017.
[54] M. G. Kibria, H. S. Kim, and I. Chong, "Tracking moving objects for intelligent IoT service provisioning in Web objects enabled IoT environment," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, 2016, pp. 561–563.
[55] O. Iova, F. Theoleyre, and T. Noel, "Using multiparent routing in rpl to increase the stability and the lifetime of the network," *Ad Hoc Netw.*, vol. 29, pp. 45–62, Jun. 2015.
[56] X. Yang, J. Guo, P. Orlik, K. Parsons, and K. Ishibashi, "Stability metric based routing protocol for low-power and lossy networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 3688–3693.
[57] O. Iova, F. Theoleyre, and T. Noel, "Stability and efficiency of RPL under realistic conditions in wireless sensor networks," in *Proc. IEEE 24th Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2013, pp. 2098–2102.
[58] A. Giyenko and Y. I. Cho, "Intelligent UAV in smart cities using IoT," in *Proc. 16th Int. Conf. Control, Autom. Syst. (ICCAS)*, 2016, pp. 207–210.
[59] Z. Khan, Z. Pervez, and A. G. Abbasi, "Towards a secure service provisioning framework in a smart city environment," *Future Generat. Comput. Syst.*, vol. 77, pp. 112–135, Dec. 2017.
[60] R. Y. Chen, "Intelligent IoT-based tracing system for backward design using FCM and fuzzy rule," in *Proc. 4th Global Congr. Intell. Syst.*, 2013, pp. 229–233.
[61] R. B. Pendor and P. P. Tasgaonkar, "An iot framework for intelligent vehicle monitoring system," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, 2016, pp. 1694–1696.
[62] J. Hou, R. Jadhav, and Z. Luo, *Optimization of Parent-Node Selection in RPL-Based Networks*, document draft-hou-roll-rpl-parent-selection-00, Internet Draft, Draft, 2017.
[63] M. M. Rathore, A. Paul, A. Ahmad, N. Chilamkurti, W.-H. Hong, and H. Seo, "Real-time secure communication for smart city in high-speed big data environment," *Future Generat. Comput. Syst.*, vol. 83, pp. 638–652, Jun. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17317557
[64] A. Aliyu *et al.*, "Towards video streaming in IoT environments: Vehicular communication perspective," *Comput. Commun.*, vol. 118, pp. 93–119, Mar. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366417305121
[65] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the Internet of Things: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 414–454, 1st Quart., 2014.
[66] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in *Handheld and Ubiquitous Computing*. Berlin, Germany: Springer, 1999, pp. 304–307.
[67] C. Perera, C. H. Liu, S. Jayawardena, and M. Chen, "A survey on Internet of Things from industrial market perspective," *IEEE Access*, vol. 2, pp. 1660–1679, Jan. 2014.
[68] S. Zairi, B. Zouari, E. Niel, and E. Dumitrescu, "Nodes self-scheduling approach for maximising wireless sensor network lifetime based on remaining energy," *IET Wireless Sensor Syst.*, vol. 2, no. 1, pp. 52–62, Mar. 2012.
[69] J. So and H. Byun, "Load-balanced opportunistic routing for duty-cycled wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1940–1955, Jul. 2017.
[70] B. Shahrokhzadeh and M. Dehghan, "A distributed game-theoretic approach for target coverage in visual sensor networks," *IEEE Sensors J.*, vol. 17, no. 22, pp. 7542–7552, Nov. 2017.
[71] A. Mtibaa, A. Fahim, K. A. Harras, and M. H. Ammar, "Towards resource sharing in mobile device clouds: Power balancing across mobile devices," *SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, pp. 51–56, 2013.
[72] P. Nayak and A. Devulapalli, "A fuzzy logic-based clustering algorithm for WSN to extend the network lifetime," *IEEE Sensors J.*, vol. 16, no. 1, pp. 137–144, Jan. 2016.

[73] G. Raina, S. Manjunath, S. Prasad, and K. Giridhar, "Stability and performance analysis of compound TCP with REM and drop-tail queue management," *IEEE/ACM Trans. Netw.*, vol. 24, no. 4, pp. 1961–1974, Aug. 2016.

[74] L. Khoshnevisan and F. R. Salmasi, "Adaptive rate-based congestion control with weighted fairness through multi-loop gradient projection internal model controller," *IET Control Theory Appl.*, vol. 9, no. 18, pp. 2641–2647, 2015.

[75] G. Carlucci, L. De Cicco, S. Holmer, and S. Mascolo, "Congestion control for web real-time communication," *IEEE/ACM Trans. Netw.*, vol. 25, no. 5, pp. 2629–2642, Oct. 2017.

[76] X. Li *et al.*, "A queue scheduling approach to QoS support in terminal communication access network," in *Proc. 12th Int. Conf. Natural Comput., Fuzzy Syst. Knowl. Discovery (ICNC-FSKD)*, 2016, pp. 1974–1979.

[77] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proc. 9th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2003, pp. 134–146.

[78] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki—A lightweight and flexible operating system for tiny networked sensors," in *Proc. 29th Annu. IEEE Int. Conf. Local Comput. Netw.*, Nov. 2004, pp. 455–462.

[79] M. Lin, H.-S. Kim, and S. Bahk, "Transmission power control for large scale industrial applications in low power and lossy networks," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2015, pp. 380–382.

SEYEDREZA TAGHIZADEH received the bachelor's degree in computer engineering in 2008 and the master's degree in information technology in 2011. He is currently pursuing the Ph.D. degree in information technology with the University of Tehran. Since 2011, he has been the Developer of security apps and IoTCloud platforms, while leading the Cloud Computing team in the Research Center of Iran Ministry of Information and Communications Technology for two years. He has also led some security projects in 2016 and 2017. His interests are mostly in network security, Internet of Things, wireless sensor networks, cloud computing, and routing protocols.

HOSSEIN BOBARSHAD received the M.Sc. degree in communication engineering from the Iran University of Science Technology, Tehran, Iran, and the Ph.D. degree in wireless networking from the Electronic Engineering Department, King's College London, London, U.K., in 2010. He was a Research Assistant with the King's College London. He is currently an Assistant Professor with the University of Tehran, Tehran, Iran. His main research interests are in the areas of data communication theory, mathematical modeling, and real-time video transmission through unreliable media, e.g., wireless networks and the Internet. In particular, he conducts his research mainly on cross-layer design for adaptive error protection in video over WLAN and wireless sensor networks for monitoring of physiological signals.

HALIMA ELBIAZE received the B.S. degree in applied mathematics from the University of MV, Morocco, in 1996, the M.Sc. degree in telecommunication systems from the Université de Versailles, in 1998, and the Ph.D. degree in computer science from the Institut National des Télécommunications, Paris, France, in 2002. Since 2003, she has been with the Department of Computer Science, Université du Québec à Montréal, QC, Canada, where she is currently a Professor. She has authored or co-authored many journal and conference papers. Her research interests include network performance evaluation, traffic engineering, and quality of service management in optical and wireless networks.