International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 8, No.6 (Nov-2019)
http://dx.doi.org/10.12785/ijcds/080611

Resource Efficient Routing in Internet of Things: Concept, Challenges, and Future Directions

Jaisooraj J[1], S D Madhu Kumar[1] and Aindrila Ghosh[1]

[1] *Computer Science and Engineering Department, National Institute of Technology Calicut, Calicut, India*

Received 31 May 2019, Revised 29 Jul. 2019, Accepted 20 Oct. 2019, Published 1 Nov. 2019

Abstract: Ever since Internet of Things (IoT) has become a reality, fresh challenges have been propping up in a day-to-day manner for keeping the paradigm robust. As an IoT enabling technology, Low Power Lossy Networks (LLNs) play a notable role in ensuring the connectivity of devices, and thereby the efficient functioning of applications. Considering their resource constrained nature, routing protocols designed for LLNs demand a proper management of vital resources. The paper considers node memory, node battery, and network bandwidth to be vital network resources, and reviews in detail the existing LLN routing protocols as well as routing metrics with respect to the level of resource management achieved by each of them. Finally, the paper enlists research challenges in the field of LLN routing, which can be considered as the need of the hour, for assisting the IoT paradigm to reach its maximum potential.

Keywords: Internet of Things, Low Power Lossy Networks, Routing, Routing Metric, Resource Efficiency

1. INTRODUCTION

Internet connectivity getting extended to everyday 'things'. In other words, 'things' communicating without human intervention. Such a sophisticated concept has become a reality through the *Internet of Things (IoT)* paradigm, the latest development in the field of *Information and Communication Technology (ICT)*. This has been made possible only through a constructive technological fusion, starting with embedded systems to attach sensors, proceeding through networking to establish connectivity among 'things'/ sensors, cloud computing to act as data centers, and finally data analytics to develop socially relevant applications. Before going into resource management issues in routing, we can have a brief look on the different layers in which the IoT paradigm operates (Fig. 1).

The Internet of Things (IoT) paradigm operates in three layers – *sensing* layer (Layer 1), *network* layer (Layer 2), and *application* layer (Layer 3). The *sensing* layer, which is also the bottommost layer, comprises of 'things' attached with sensors. These sensors form a network of its own, which extends up to an IoT gateway. Such a network is often referred to as *Low Power Lossy Networks (LLNs)*. *Network* layer handles the connectivity among such IoT gateways spanning across different LLNs. The uppermost layer, i.e. the *application* layer is responsible for performing analytics on sensed data. Any IoT application, let it be smart city, smart home, healthcare, agriculture, or industry, persistent connectivity among sensors in the sensing layer is required. Obviously, routing protocols in Low Power Lossy Networks (LLNs) play an important role in ensuring such a pervasive connectivity.

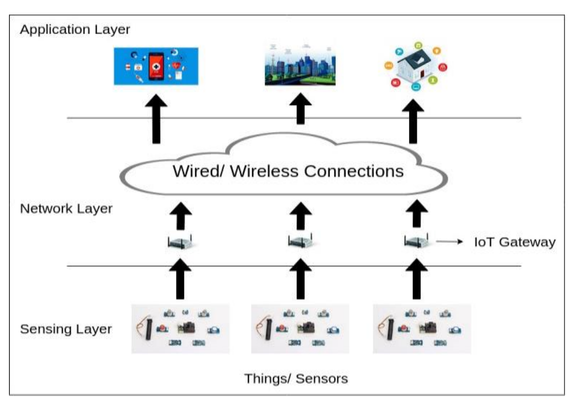

Figure 1. Layers in the IoT paradigm.

Low Power Lossy Networks (LLNs) have been categorized as resource constrained networks, by the *Internet Engineering Task Force (IETF)* [1]. Constrained

E-mail address: jaisooraj11@gmail.com, madhu@nitc.ac.in, aindrila22@gmail.com

nodes form a significant portion of constrained networks. Specific characteristics, like battery power, memory, and bandwidth, which are taken to be granted for normal Internet nodes, are not achievable for constrained nodes. Consequently, the routing protocols employed on LLNs should take into account, the efficient management of these resources. Inefficient handling of such resources can result in frequent connectivity disruptions, and thereby affect the performance of higher layers, including the application layer. Considering these aspects, the IETF has created a working group named *Routing Over Low Power Lossy Networks (ROLL)* to design and develop efficient routing protocols for LLNs.

The first and foremost contribution of ROLL, towards LLN routing, is the development of *Routing Protocol for LLNs (RPL)* [2]. Often referred to as the de facto routing protocol for IoT, RPL operates in a proactive manner, taking into account the resource constrained nature of LLN nodes. In order to keep up with the rapid technological advancements as well as to meet the ever increasing demand for IoT applications, several enhancements have been proposed over the core RPL protocol. Another contribution from ROLL is an on-demand routing protocol for LLNs, i.e., *Lightweight On-demand Ad hoc Distance vector routing protocol – next generation (LOADng)* [3]. In addition to RPL and LOADng, several opportunistic routing protocols have also been developed by eminent researchers. Several useful surveys/ reviews have been undertaken by the researchers in the field of LLN routing, be it in point-to-point communications [4], mobility [5], or security [6]. Still, there is a noticeable lack of a review which takes into consideration the resource management issues in LLNs.

The paper presents a detailed review on the three main categories of LLN routing namely, proactive, on-demand, and opportunistic routing. The review has been conducted from the viewpoint of degree to which resource management has been achieved by various protocols, with the resources considered being node memory, battery capacity of nodes, and network bandwidth. Also, various LLN routing metrics have been studied based on their role in resource management. Further, the paper enlists important challenges that need to be addressed for developing more efficient LLN routing protocols, so that the needs of a future smart world can be met.

Remainder of the paper is organized as follows: Section 2 provides a detailed review of the existing LLN routing protocols, Section 3 provides a quantitative analysis of LLN routing protocols, Section 4 reviews the various LLN routing metrics, Section 5 offers a list of open challenges in the field of LLN routing, and Section 6 concludes the paper.

2. RELATED WORK

Routing Over Low Power Lossy Networks (ROLL) working group has been rightly credited with the design of both proactive as well as on-demand LLN routing protocols. RPL and its variants comprise the family of proactive routing protocols, whereas LOADng along with its variants constitute the category of on-demand routing protocols. These two categories of routing, along with opportunistic routing constitutes the body of literature regarding LLN routing. This section presents a detailed analysis of protocols belonging to each of the three categories of routing. Proactive routing protocols have been analyzed under RPL, whereas on-demand routing protocols come under LOADng. These two classes, along with opportunistic routing will be discussed in the coming subsections. The hierarchy of such a classification has been provided in Fig. 2.

A. Routing Protocol for Low Power Lossy Networks (RPL)

Ever since its inception in 2012 by the Internet Engineering Task Force (IETF) [2], RPL has maintained its title as the most researched entity in the family of LLN routing protocols. Being a proactive/ table-driven routing protocol, RPL bypasses the condition of an increased control overhead, thereby saving network resources like battery power, node memory, and even bandwidth. RPL takes into consideration all the three kinds of traffic – point-to-multipoint (or gateway to nodes), multipoint-to-point (or nodes to gateway), and point-to-point (or device to device).

RPL operates by constructing a *Destination Oriented Directed Acyclic Graph (DODAG)* pointed towards the root. The root node can either be a gateway, or it can even be a node connected to the gateway. In either case, the root node is not resource constrained, and is connected to a constant power supply. The remaining nodes in DODAG characterizes 'things' which are constrained in nature. The topology construction is initiated by the root node, by transmitting a control message named *DODAG Information Object (DIO)*. This DIO message, received by all the one hop neighbors of root, consists of the rank of transmitter/parent (in this case, the root itself with rank set to one) along with the *Objective Function (OF)* used to compute the rank. It may be noted here that, each and every node present in a specific DODAG should adhere to the same OF. The receiving nodes calculate their rank using OF (the OF takes into consideration, rank of the parent). These nodes replace the rank field in DIO with their newly computed rank, and broadcasts it. Whenever a node receives DIO messages from multiple parents, the parent with minimum rank is chosen as the preferred one. After such a construction of the DODAG topology, DIO messages are transmitted periodically by each node. This periodic DIO transmission, governed by a trickle timer, is used to maintain the topology under dynamic network conditions. Apart from DIO, *DODAG Information Solicitation (DIS)* and *Destination Advertisement Object (DAO)* form the other important control messages. DIS facilitates the spontaneous transmission of DIO messages, thereby allowing new/ disconnected nodes to get attached/ reattached to the topology. DAO messages, on the other hand, express a node's willingness to act as destination. In more technical terms, DAO messages handle point-to-multipoint communications, i.e., network traffic from gateway (or root) to LLN nodes.

Even though RPL remains as the primary IoT routing protocol, significant enhancements need to be made on top of RPL in order to keep up with the technical and social demands of IoT paradigm. The following subsections provide an in-depth review of the various enhancements made on RPL. For convenience, the works have been classified according to the type of enhancement made. Keeping the aim of this paper intact, the review has been performed from the view of resource efficiency achieved by each of the protocols.

a) Based on the Modes of Operation: RPL operates in two modes – non-storing mode and storing mode. Non-storing mode requires only the root to store the routing information, whereas the storing mode of operation requires each and every node in the LLN to store routing information. TABLE I and TABLE II shows the routing tables of non-storing and storing modes respectively, as per the network topology shown in Fig. 3. Each of the above two modes comes with its own share of pros and cons. For instance, the fact that non-storing mode doesn't require individual nodes to store routing information, ensures that the resources like node memory and battery are managed. But, it has a disadvantage that network traffic surrounding the root will increase. For instance, even if the source and destination nodes are siblings as per topology, packets will have to traverse the root. This can prove to be extremely inefficient as it can waste the network bandwidth of area surrounding the root, and can drain considerable battery capacity of nodes in that area. Coming to the storing mode of operation, the avoidance of such a condition comes at the expense of memory capacity of nodes. Consequently, an efficient mixing of these two modes of operation in the same network is needed. This is contrary to the IETF version of RPL, where all the nodes in a single network are either storing or non-storing.

Notable enhancements on RPL have been developed on the basis of an enquiry: can we have both storing and non-storing nodes in the same network? J. Guo in [7] proposes an enhanced version of RPL, in the name *RAM-RPL (Resource-aware Adaptive Mode RPL)*, which allows a resource aware adaptive switching of operating modes. Consequently, this allows resources like battery power and memory to be managed effectively. In [8], the authors present a Dual-MOP RPL, which allows for an effective management of storing and non-storing nodes while considering downward traffic. Similar to [7], battery capacity and node memory are managed efficiently in [8] also. But, both these approaches are not that successful in the effective utilization of network bandwidth. This is because of the use of a combined approach involving both source routing and hop-by-hop routing. Reference [9] takes into consideration the fact that a network with predominantly storing nodes can pose serious scalability issues, and puts forward an enhanced version of RPL named D-RPL. Here, multicast group messages have been employed so as to avoid memory overflow in nodes. Even though memory is handled up to an extent in this work, it can result in considerable wastage of bandwidth surrounding the root. Further, it can drain out the battery of nodes surrounding the root. Another important contribution put forward by K. Vyas in [10], termed *Adaptive RPL (ARPL)*, permits the storing nodes to get converted into non-storing mode on an event of routing table overflow. Hence, it is obvious to conclude that the node memory is optimized. Also, it reduces the traffic surrounding the root, thereby reducing the bandwidth utilization. But, residual battery, which is an important aspect in a routing metric, has not been considered while performing the switch. TABLE III shows a mapping between such RPL variants and the features that enable them to manage resources.

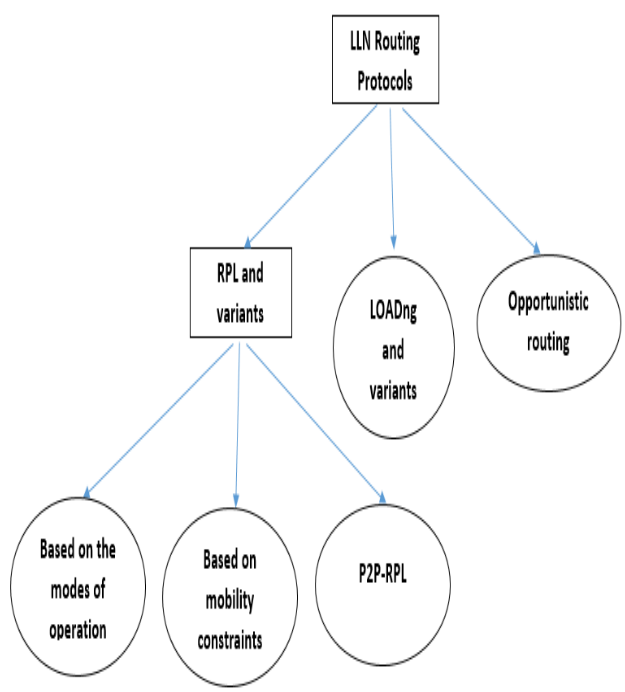

Figure 2. Hierarchy of LLN routing

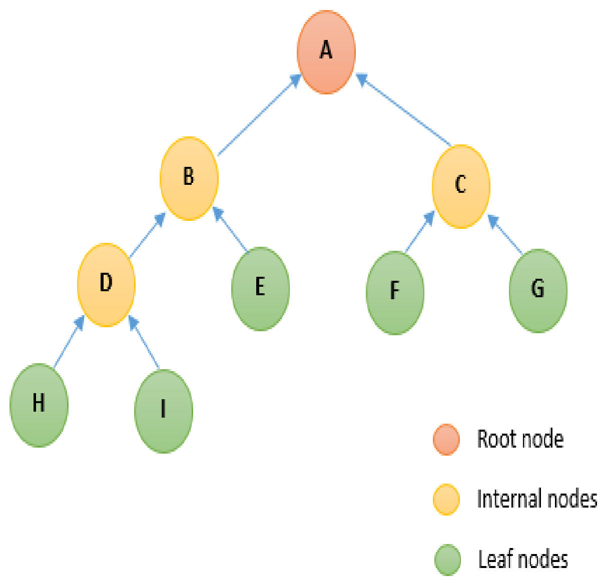

Figure 3. An example LLN topology

TABLE I. ROUTING TABLE FOR NON-STORING MODE

Node	Parent	Routing Entry (Destination, Parent)
A	Nil	(B, A), (C, A), (D, B), (E, B), (F, C), (G, C), (H, D), (I, D)
B	A	Nil
C	A	Nil
D	B	Nil
E, F, G, H, I	B, C, C, D, D	Nil

TABLE II. ROUTING TABLE FOR STORING MODE

Node	Parent	Routing Entry (Destination, Parent)
A	Nil	(B, A), (C, A), (D, B), (E, B), (F, C), (G, C), (H, D), (I, D)
B	A	(D, B), (E, B), (H, D), (I, D)
C	A	(F, C), (G, C)
D	B	(H, D), (I, D)
E, F, G, H, I	B, C, C, D, D	Nil

b) Based on Mobility Constraints: As the scale and demand of Internet of Things (IoT) paradigm have gone up (and still continues to go up), mobility has become a criterion that can't afford to be missed. With the advent of applications like Industrial IoT (IIoT), RPL which was actually developed for static LLNs, had to accommodate mobility. Contrary to the enhancements made on the modes of operation, which mainly focus on node memory, mobility enhancements are made primarily for handling bandwidth and node battery. Also, it won't be incorrect to state that the majority of works being taken up in this area, are to handle mobility issues in RPL. Resource management achieved by such RPL variants has been summarized in TABLE IV.

The authors in [11] developed a variant of RPL where the trickle timer is set values ranging from I_{min} to I_{max}. This ensures that the control overhead is reduced in the network, thereby preserving network bandwidth. However, the paper does not ensure a proper management of node memory and battery. In addition to that, larger trickle timer values can lead to frequent connectivity disruptions. Another important contribution, towards ensuring mobility support for RPL, proposes a reverse trickle timer for mobile nodes [12]. It is based on the observation that a mobile node is likely to stay connected with its new preferred parent for a long time, and hence the interval for sending DIO messages will be high at the beginning. Even though it manages the network bandwidth by reducing the number of control messages, the base assumption might not always be true. Fotouhi et al. proposed an enhancement in the name of MRPL, which employs the concept of smart hops used in cellular networks [13]. Whenever the strength of a link between mobile node and its parent goes below a threshold, such a connection is immediately terminated. Then, the mobile nodes goes into a parent discovery process by sending DIS messages. In this enhancement, node battery is efficiently handled through parent switching. But, a frequent transmission of DIS messages can waste the network bandwidth. The same authors proposed an enhancement to MRPL in the name MRPL+ [14]. Here, the disconnection with a parent happens only after finding a potential new parent. Even though the connectivity issues are addressed in [14], the issue of bandwidth management remains unaddressed. Sanshi et al. in [15] puts forward a protocol named *Enhanced RPL (ERPL)*, which uses different Objective Function (OFs) for static and mobile nodes. Such an enhancement helps in preserving node battery as well as network bandwidth. The *Backpressure RPL (BRPL)* proposed in [16] uses QuickTheta and QuickBeta algorithms for handling diverse traffic patterns and mobility respectively. This allows for the effective management of bandwidth as well as residual battery, but not node memory. The same can be said regarding *Game Theory based Mobile RPL (GTM-RPL)* [17], which uses game theory based optimization. As an extension to [15], Sanshi et al. proposed an *Enhanced Mobility RPL (EM-RPL)* in [18], which uses various timers to keep track of mobility in

LLNs. But here, bandwidth is not optimized as effectively as the battery capacity of nodes.

c) Point-to-Point RPL (P2P-RPL): Even though the communications happening within the IoT paradigm span across multiple networks, we cannot skip the case where both source and destination nodes are present in the same network (i.e., LLN). One can simply intuit such kind of a communication in the context of smart homes or smart hospitals. Remote control based applications, in the above scenarios, denote worthy examples of such a communication. The devices present in the same network communicating with each other, has been identified by the term point-to-point communication or device-to-device communication. Despite the ability of RPL to handle such communications, bandwidth and battery wastage induced by it needs rectification.

TABLE III. RESOURCE MANAGEMENT IN RPL VARIANTS BASED ON OPERATING MODES ('**X**' INDICATES NO SPECIFIC MEASURES TO MANAGE THAT RESOURCE)

Protocol	Node Memory	Node Battery	Network Bandwidth
RAM-RPL [7]	• Resource aware adaptive switching between storing and non-storing modes	• Resource aware adaptive switching between storing and non-storing modes	• Less traffic surrounding the root, due to the presence of storing nodes
Dual-MOP-RPL [8]	• Switching of nodes among storing and non-storing modes	• Switching of nodes among storing and non-storing modes	• Optimization of downlink traffic
D-RPL [9]	• Less number of storing nodes	• Less number of storing nodes	**X**
ARPL [10]	• Switching from storing to non-storing in case of routing table overflow	**X**	**X**

TABLE IV. RESOURCE MANAGEMENT IN RPL VARIANTS BASED ON MOBILITY CONSTRAINTS ('**X**' INDICATES NO SPECIFIC MEASURES TO MANAGE THAT RESOURCE)

Protocol	Node Memory	Node Battery	Network Bandwidth
I_{min}, I_{max} [11]	**X**	**X**	• Adaptive trickle timer values
Reverse Trickle [12]	**X**	**X**	• Usage of reverse trickle timer for mobile nodes
MRPL [13]	**X**	• Usage of smart hops for preferred parent switching	**X**
MRPL+ [14]	**X**	• Usage of smart hops for preferred parent switching	**X**
ERPL [15]	**X**	• Different OFs for static and mobile nodes	• Different OFs for static and mobile nodes
BRPL [16]	**X**	• QuickTheta algorithm • QuickBeta algorithm	• QuickTheta algorithm • QuickBeta algorithm
GTM-RPL [17]	**X**	• Payoff function which considers utility, mobility, energy, and priority of nodes	• Payoff function which considers utility, mobility, energy, and priority of nodes
EM-RPL [18]	**X**	• Different OFs for static and mobile nodes • Adaptive timers	• Different OFs for static and mobile nodes

As already mentioned in this section, RPL handles point-to-point communications. It utilizes DODAG to handle the same. Now, we have to look separately from the non-storing viewpoint as well as the storing viewpoint. A non-storing view points to the fact that each and every communication has to pass through the root, even if the source and destination nodes are in close proximity. This can result in serious bandwidth as well as battery wastage surrounding the root. On the other hand, storing point of view ensures that such a condition is eliminated, but at the cost of memory capacity of nodes. All these have prompted the IETF to draft another RPL enhancement, named *Point-to-Point RPL (P2P-RPL)* [19].

P2P-RPL operates through an on-demand creation of DODAGs for device-to-device communications. When a node wishes to communicate with another node in the same network, it sends a DIO message piggybacked with a *P2P Route Discovery Option (P2P-RDO)*. This initializes the construction of an on-demand DODAG rooted at the source node. Until the P2P-RDO reaches the intended destination, the procedure of parent selection is similar to the one followed in RPL. The destination, on receiving the P2P-RDO, replies with a *P2P Discovery Reply Object (P2P-DRO)* through the discovered route. Once the P2P-DRO message has reached the source node, actual data transmission begins through the established route. Although P2P-RPL tackles the issue of an increased traffic specifically surrounding the root, it induces some additional overhead as in the case of any on-demand routing mechanism. As a countermeasure, [20] and [21] utilize a mix of RPL and P2P-RPL in the same network. Here, the on-demand DODAG discovery is initiated only if the cost of route in pre-existing DODAG (i.e., using RPL) is greater than a threshold. Such a route cost is calculated using a *Measurement Object (MO)*, explained in [20]. Such a measure can reduce the overhead associated with a purely on-demand procedure, consequently improving the bandwidth utilization. TABLE V shows the resource management capabilities of point-to-point routing protocols.

B. Lightweight On-demand Ad hoc Distance vector routing protocol – next generation (LOADng)

Ever since the advent of *Mobile Ad hoc Network (MANET)* working group by the IETF, the development of on-demand routing protocols have gained momentum. Such a category of routing protocols, which set up routes only on need, carries with it an additional overhead of increased control messages. When we are considering LLNs, which are resource constrained in nature, an increased control overhead can adversely affect the resources like network bandwidth as well as battery. It was only after taking into consideration all these aspects, that the IETF working group named *6LoWPAN (IPv6 over Low power WPAN)* developed an on-demand routing protocol called *LOAD (6LoWPAN Ad Hoc On-Demand Distance Vector Routing)* [22]. However, the development of LOAD suffered an eventual suspension as a consequence of the group giving more emphasis on issues like IP packet header compression [22]. Once again, a stable and significant development of an on-demand protocol for LLNs was made by ROLL in 2013, with the LOADng protocol [3].

LOADng, as is the case for all on-demand protocols, relies on *Route Request (RREQ)* and *Route Reply (RREP)* messages for route discovery. When a node wishes to communicate with another, it performs a link based multicast of the RREQ message. The destination, on receiving the RREQ, replies by sending an RREP message through a unicast towards the source. Through the route thereby established, the source-destination pair can communicate. One can surely raise an issue that these are features present in already developed on-demand protocols. What makes LOADng so peculiar? The answer is that the route discovery as well as maintenance phases are highly simplified, so as to suit the requirements of a resource constrained network (as in the case of LLNs). First of all, intermediate route reply, i.e. the intermediate nodes sending RREP when they have a working route towards the destination, is disallowed in LOADng. In addition to that, LOADng is extensible, so that additional features can be incorporated to the core protocol. This is especially important, considering the extensible nature of IoT paradigm. Another unique feature of LOADng is the support for multiple metrics, which is also essential considering the multiple *Quality of Service (QoS)* requirements in LLNs. Finally, the fact that LOADng supports address lengths ranging from 1 to 16 octets is quite advantageous for the memory constrained LLN nodes.

Even in the midst of all these desirable features, the control overhead caused by LOADng needs due consideration. Important extensions on LOADng protocol have been explained very clearly in [22]. TABLE VI outlines the resource management achieved by various LOADng extensions. A brief account on these extensions has been given below:

- **Smart Route Request** – The usage of SmartRREQ is aimed at avoiding the unwanted broadcast of RREQ messages, thereby saving a lot of network bandwidth. In this extension, RREQ is broadcast with a flag set so as to communicate its smart nature. When an intermediate node receives such an RREQ (or smartRREQ), it checks for the presence of a route towards the intended destination, in its routing table. In order to avoid a routing loop, it is made sure that the next hop of that route is not the same as previous hop of the RREQ. If both of these conditions are met, the RREQ is unicast towards the next hop. Otherwise, it is broadcasted as is the case with normal LOADng. Also, such an extension satisfies the interoperability requirements, so as to operate along with nodes running on base LOADng protocol.

TABLE V. RESOURCE MANAGEMENT IN POINT-TO-POINT ROUTING PROTOCOLS ('X' INDICATES NO SPECIFIC MEASURES TO MANAGE THAT RESOURCE)

Protocol	Node Memory	Node Battery	Network Bandwidth
P2P-RPL [19]	X	• On-demand DODAG construction • Reduced traffic surrounding the root	• On-demand DODAG construction • Reduced traffic surrounding the root
RPL & P2P-RPL [20]	X	• On-demand DODAG construction • Reduced traffic surrounding the root	• On-demand DODAG construction • Reduced traffic surrounding the root • Adaptive switching between RPL and P2P-RPL

TABLE VI. RESOURCE MANGEMENT IN LOADNG VARIANTS ('X' INDICATES NO SPECIFIC MEASURES TO MANAGE THAT RESOURCE)

Protocol	Node Memory	Node Battery	Network Bandwidth
smartRREQ [22]	• Varying address lengths from 1 to 16 octets	• Avoidance of unwanted RREQ flooding	• Avoidance of unwanted RREQ flooding
Expanding Ring [22]	• Varying address lengths from 1 to 16 octets	• Avoidance of unwanted RREQ flooding through step-by-step increase in TTL	• Avoidance of unwanted RREQ flooding through step-by-step increase in TTL
LOADng-CTP [22]	X	X	• Appropriate usage of jitter values
LOADng-DFF [22]	X	X	• Usage of Depth First Forwarding
LOADng-DFF++ [22]	• Timely rearrangement of CNHL	X	• Usage of Depth First Forwarding

- **Expanding Ring** – Once again, the aim is to conserve network bandwidth by limiting the broadcast of RREQ messages. In the expanding ring version of LOADng, this is achieved by periodically increasing the *Time-to-Live (TTL)* value of RREQ messages. The initial iteration of this extension broadcasts the RREQ messages with a reduced TTL value. If the RREQ is not received at the intended destination, the whole process is restarted at the source with an increased TTL value. Such an approach has the advantage that an RREQ message traverses the entire network only in the worst case. Here also, interoperability with basic LOADng is achieved. Nodes running on basic LOADng won't be able to recognize the extension, and hence it won't reduce the TTL value.
- **Collection Trees** – This extension, termed as *LOADng-Collection Tree Protocol (LOADng-CTP)*, is aimed at handling multipoint-to-point communications. Here, the node which intends to act as the root broadcasts an RREQ_Trigger message so as to set up a collection tree with each node having bidirectional links. All the messages required for setting bidirectional links, be it either RREQ_Trigger or HELLO, are sent by considering a suitable jitter. This allows to avoid unwanted collisions. Once the collection tree is set up, all the nodes in the tree will have a valid bidirectional route towards the root. Thus, we can

safely presume that LOADng-CTP manages a lot of network bandwidth while handling multipoint-to-point communications. Unlike the previous two extensions, LOADng-CTP cannot interoperate with LOADng routers. This is on account of the inability of LOADng routers in generating HELLO messages, which in turn is crucial for establishing bidirectional links.

- **Depth First Forwarding (DFF)** – LOADng-DFF offers an advantage of continuing the data transmission even if the next hop to a particular destination fails. This is achieved by maintaining a *Candidate Next Hop List (CNHL)*, i.e. neighbors with bidirectional links, for each node. First element in the list will be the one suggested as next hop for a specific destination in the routing table. The remaining elements are those discovered through *NHDP (Neighborhood Discovery Protocol)*, not following any specific sequence. Whenever a data packet reaches the node, it gets forwarded to the first element in its CNHL. If the routing fails for some reason, the packet is forwarded to the next element in CNHL. Thus, LOADng-DFF ensures a persistent connectivity, thereby leading to an efficient utilization of bandwidth. However, it does not rearrange the elements in CNHL as per the success rate. This can lead to an unwanted delay in the subsequent transmissions passing through that node. This, along with the lack of interoperability with LOADng, can be said as the drawbacks of LOADng-DFF.
- **Depth First Forwarding++ (DFF++)** – LOADng-DFF++ tackles the above disadvantage (of LOADng-DFF) regarding ordering of elements in CNHL. Here there is a requirement that the particular neighbor (in CNHL) through which a successful transmission to the destination was possible, needs to be appended at the end of the list. This ensures that the subsequent transmissions through that node, will first choose the previously successful CNHL element as the next hop, thereby reducing considerable delay as well as overhead. Even though LOADng-DFF++ interoperates with LOADng-DFF, backward compatibility with LOADng is not ensured.

The above extensions made on top of LOADng have surely helped to reduce the inherent control overhead and end-to-end delay, and thereby optimize the network resources especially network bandwidth and node battery. However, further enhancements, like multipath routing, can be incorporated so as to further optimize its performance. Also, the fact that LOADng-DFF and LOADng-DFF++ are not interoperable with LOADng needs due consideration. As the future technical world is ready to witness a massive growth of IoT, development of optimized on-demand protocols occupies a place that cannot be left out.

C. Opportunistic Routing

Wireless network researchers will surely be familiar of the term *Delay/ Disruption Tolerant Networks (DTNs)*. DTNs actually came into being as an offshoot of MANETs. In fact, they were often addressed as *IC-MANETs (Intermittently Connected MANETs)*. We are already familiar with the fact that MANETs possess a highly dynamic networking topology, thereby making the routing procedure extremely complex (NP-Hard if we think from a theoretical perspective). However, we can set up routes for communication in a MANET, even though they are unstable. Coming to DTNs, we can't even setup routes. Instead, they rely on opportunistic contacts in order to transfer messages, so that the messages may eventually reach the destination. That is the reason for DTNs to be called by the name opportunistic networks. One may wonder whether such a high degree of mobility occurs in IoT. The answer is that, considering the rate at which IoT paradigm is advancing, such a level of dynamicity is quite possible in the near future itself. So, we can consider opportunistic LLN routing protocols in three contexts, which are detailed out below.

The first context, where opportunistic routing comes handy, is closely related to Industrial IoT (IIoT) as well as vehicular IoT. For instance, we can consider a scenario where roadside sensors/ actuators have to connect with moving devices (these devices could also be present in moving vehicles). Such a communication is possible only if the roadside devices are able to opportunistically connect with their moving counterparts. Such a scenario is given the term *opportunistic IoT (oppIoT)* [23] [24]. Considering the fact that the time to make opportunistic contacts is way too less, the design of such routing protocols is a very complex task.

The other context involves an opportunistic network connected to the Internet, i.e., LLN assumes the nature of an opportunistic network/ DTN. Here, the opportunistic routing technique of store-carry-forward needs to be used. A source node carries the message to be transmitted in its buffer, until it comes into contact with another node. The source makes use of such an opportunistic contact to transfer the message. This process is continued until the message reaches the intended destination. The authors, in [24], proposes such a routing scheme, where they make use of *Location Prediction based Forwarding for Routing using Markov Chain (LPFR-MC)* in order to figure out the subsequent positions of each node. Such a method, which takes into consideration the angular metrics of nodes, is useful in improving the frequency of opportunistic contacts. This can prevent the data packets getting congested in the network due to lack of such contacts, and hence it can save considerable network bandwidth.

Last but surely not the least, the third context is the one where comparatively more amount of research have been done in opportunistic LLN routing. Here, the opportunistic routing is incorporated into RPL. *Opportunistic RPL (ORPL)* is first in this category, where a routing table free approach is used [25]. A node can decide its preferred parent opportunistically. As a result, considerable node memory is saved here. An enhancement to ORPL has been provided in [26] in the name of *ORPL-LB (ORPL-Load Balanced)*. Here, duty cycling has been introduced so as to balance the network load uniformly among all nodes in the network. This enables the management of both node battery as well as network bandwidth. Gormus et al., in [27] and [28], tests the performance of ORPL on top of *Adaptive Metering Infrastructure (AMI)* mesh networks. In [28], an enhancement named ORPLx has been proposed by the authors, where an adaptive limit is put on medium access control retransmissions. Consequently, network bandwidth is saved through the reduction of control overhead. Similar to previous tables presented in this paper, TABLE VII provides a mapping between resource management and the enabling features for opportunistic routing protocols.

Compared to its proactive and reactive counterparts, opportunistic routing in LLNs has a less number of takers. This could be due to the fact that the evolution of IoT has not yet reached a stage suited for this category of protocols. Also, the end-to-end delay resulted by opportunistic routing protocols is something to addressed. However, with terms like *Internet of Drones (IoD)* and *Flying Ad hoc Networks (FANETs)* becoming popular, no one can predict the rate at which the IoT paradigm will advance. Hence, it would be better to develop efficient opportunistic protocols before the well expected IoT rush culminates itself into a highly complex scenario.

3. QUANTITATIVE ANALYSIS OF LLN ROUTING

In Section 2, we have already discussed a qualitative analysis regarding the various categories of LLN routing protocols. There, we had given importance to the specific characteristics of individual protocols that enable them to manage resources such as battery, memory, and network bandwidth. However, an analysis on LLN routing is complete only with some insights on its quantitative aspects. Coming to the quantitative aspects, packet delivery ratio, control overhead, and end-to-end delay constitute the major players. Packet delivery ratio measures the number of packets successfully received at the destination, out of the total number of packets sent. Control overhead refers to the total number of control messages required to transmit a unit of data, whereas end-to-end delay indicates total time elapsed between a successful send and receive operations. An ideal routing protocol, not only in the case of LLNs, should guarantee an increased packet delivery ratio together with a reduced control overhead and end-to-end delay. Table VIII gives a summary regarding the level of packet delivery ratio, control overhead, and end-to-end delay achieved by different LLN routing protocols, together with an indication on battery consumption.

As evident from Table VIII, on-demand and opportunistic LLN routing protocols tend to give better packet delivery ratios as compared to their proactive counterparts. However, proactive routing protocols show considerably reduced control overhead and end-to-end delay. Hence, further research is essential to develop proactive LLN routing protocols with better packet delivery ratios. Similarly, there should also be research attempts at reducing the overhead and delay associated with on-demand and opportunistic protocols.

4. LLN ROUTING METRICS

Similar to the case in any kind of a network, LLNs also take into account various metrics that govern the routing decisions. The objective of using these metrics can range from the selection of preferred parent in RPL to ranking of routes in LOADng. The routing metric is often mapped using an optimization function named Objective Function (OF), which we have already seen in Section 2. An OF can be either a minimization or a maximization function as per the network/ routing requirements. These are all general aspects that are applicable to any kind of a network. Coming to LLNs, these routing metrics should take into consideration the inherent features of a resource constrained network. This could either be battery capacity of nodes, speed of nodes, transmission counts, or even a combination of two or more of the individual metrics [29] [30]. The proper design of an LLN routing metric can cover a significant distance in the management of vital network resources. TABLE IX shows the degree of resource management that can be achieved by different routing metrics. Given below is a list of the most important LLN routing metrics available in the literature:

- **Expected Transmission Count (ETX)** – One could confidently argue that ETX is the most frequently used routing metric in LLNs. ETX indicates the number of link layer transmissions for a node to transfer a packet successfully to the destination. Naturally, the aim will be to select links with a reduced ETX value. Consequently, the OF corresponding to ETX will be a minimization function. From the viewpoint of resource management, ETX can handle bandwidth utilization effectively.

- **Link Quality Index (LQI)** – A metric, introduced along with the IEEE 802.15.4 standard, which measures error in the modulation of successfully received packets [31]. This takes into account several factors like the distance between source and destination, mobility etc. It is always favorable to have a reduced LQI value, once again pointing towards an OF which is a minimization function. LQI gives a fairly good detail regarding the bandwidth utilization.

- **Received Signal Strength Indicator (RSSI)** – As the name itself implies, RSSI indicates the power level which a receiver experiences from the sender. Factors like distance, mobility, and node battery can affect the RSSI value. Contrary to ETX and LQI, it is better to have an increased RSSI value. As a result, the OF used here will be a maximization function. Coming to resource management, RSSI can manage node battery and network bandwidth effectively.

- **Link Stability Metrics** – These can be considered as composite, rather than standalone, metrics. The stability of a link depends on the characteristic of nodes situated at both the endpoints. In addition to that, metrics like ETX, LQI, and RSSI can also affect the link characteristics. A properly designed link stability metric efficiently combines various individual metrics by assigning appropriate priorities/ weightages to each of them. A composite metric thus designed can be useful in managing all the important network resources.

TABLE VII. RESOURCE MANAGEMENT IN OPPORTUNISTIC PROTOCOLS ('**X**' INDICATES NO SPECIFIC MEASURES TO MANAGE THAT RESOURCE)

Protocol	Node Memory	Node Battery	Network Bandwidth
LPFR-MC [24]	• Store-carry-forward	**X**	• Usage of Markov Chain model to predict the location of nodes
ORPL [25]	• Routing table free	• No need to compute rank for parent selection	• Opportunistic preferred parent selection
ORPL-LB [26]	• Routing table free	• Load balancing • Duty cycling	• Load balancing • Duty cycling
ORPLx [28]	• Routing table free	• No need to compute rank for parent selection • Adaptive limit on MAC retransmissions	• Opportunistic preferred parent selection • Adaptive limit on MAC retransmissions

TABLE VIII. QUANTITATIVE ANALYSIS OF LLN ROUTING PROTOCOLS

Protocol	Packet Delivery Ratio	Control Overhead	End-to-End Delay	Battery Consumption
RAM-RPL [7]	Medium	Low	Low	Medium
Dual-MOP-RPL [8]	High	Medium	Medium	Medium
D-RPL [9]	High	Low	Medium	Medium
A-RPL [10]	High	Low	Medium	Medium
I_{min}, I_{max} [11]	Medium	Medium	Medium	Medium
Reverse Trickle [12]	Medium	Medium	Medium	Medium
MRPL [13]	High	Medium	Medium	Medium
MRPL+ [14]	High	Medium	Medium	Medium
ERPL [15]	Medium	Medium	Medium	Low
BRPL [16]	High	Medium	Medium	Low
GTM-RPL [17]	Medium	Medium	Low	Low
EM-RPL [18]	Medium	Low	Medium	Low
P2P-RPL [19]	Medium	Medium	Medium	Low
RPL & P2P-RPL [20]	Medium	Medium	Low	Low
smartRREQ [22]	High	Medium	Medium	Low
Expanding Ring [22]	High	Medium	Medium	Low
LOADng-CTP [22]	Medium	High	Medium	Medium

LOADng-DFF [22]	Medium	High	Medium	High
LOADng-DFF++ [22]	High	Medium	Medium	Medium
LPFR-MC [24]	Medium	Medium	Medium	Medium
ORPL [25]	Medium	Medium	Medium	Medium
ORPL-LB [26]	High	Medium	Medium	Low
ORPLx [28]	High	Low	Medium	Low

TABLE IX. ROUTING METRICS VS LEVEL OF RESOURCE MANAGEMENT

Routing Metric	Node Memory	Node Battery	Network Bandwidth
ETX	Low	Medium	High
LQI	Low	Medium	High
RSSI	Low	High	High
Link Stability Metrics	High	High	High

So far, we have seen metrics which are primarily confined to a link. In the case of protocols like RPL, this might work well, considering the fact that operations like preferred parent selection are concerned with links rather than routes. But in the case of LOADng, where routes have to be considered as a whole, we need to be aware of techniques to combine individual link metrics. Summation of individual link values is one among such methods to find out the value corresponding to an entire route. Even more advantageous will be a max-min approach. Finding out the link with minimum value in each route, is followed by obtaining the maximum of those values. The route possessing such a max-min value is considered to be the optimal one.

LLNs, being an integral part of the IoT, consists of nodes demanding multiple Quality of Service (QoS) requirements. Also, multiple applications may coexist in the same LLN with each having its own specific QoS requirements. Hence, the design of routing metrics taking into consideration all these aspects is mandatory. In addition to that, considering the number of potential IoT applications waiting to get realized, further research is absolutely necessary in the design of routing metrics.

5. Open Challenges and Recommendations

Considerable amount research already undertaken in the field of LLN routing doesn't mean that the space for further research has been wiped out. Instead, it won't be wrong even if we go an extend stating that the space for further research is more, compared to that of the research already carried out. The major research challenges, as per the review conducted, along with possible solutions have been explained below:

- **Scalability** – This comes foremost among the list of challenges in LLN routing. With Internet of Things (IoT) being a paradigm for making 'things' a part of the global Internet infrastructure, there is no limit regarding the number of 'things' which would get connected in the future, or say near future. Scalability, defined as the ability to prevent performance degradation even if the network grows in size, comes handy in such a scenario.

  Enforcing scalability requires measures unique to each category of routing. For instance, the scalability of RPL can be enhanced by carrying out a proper switch between storing and non-storing modes of operation. If the number of non-storing nodes increase, it will result in an increased traffic surrounding the root. Such an increase in traffic is directly proportional to the network size (i.e. number of nodes in the network), as the non-storing nodes will simply forward the traffic towards the root. Now coming to LOADng, more number of nodes would mean an increased amount of control messages being transmitted. Hence, measures should be taken to counter that. There is a difference in the case of opportunistic routing though, as the frequency of opportunistic contacts may increase with the number of nodes.

- **Mobility** – The rapid evolution of IoT paradigm has led to a condition where mobility is of utmost importance while designing an LLN routing protocol. The 'things'/ sensors are being increasingly carried by people or even vehicles. As a result, routing protocols that do not take into consideration the mobility requirements can suffer from frequent loss of connectivity.

  The impact of mobility is also distinct for various classes of routing protocols. RPL, originally designed for static/ less mobile networks, has seen a lot of enhancements so as to accommodate mobility. But, such enhancements are quite difficult considering the proactive nature of RPL. Even though LOADng and opportunistic routing protocols support mobility, it comes with the cost of an increased control overhead and end-to-end delay respectively.

- **Bidirectional Traffic** – This is the characteristic feature of LLNs, which distinguishes them from traditional *Wireless Sensor Networks (WSNs)*. Unlike WSNs, LLNs are connected to the Internet. This ensures both uplink as well as downlink traffic, i.e., multipoint-to-point and point-to-multipoint traffic. For a routing protocol to be completely fit in an LLN environment, both these kinds of traffic should be supported.

  Considering the case of RPL, multipoint-to-point traffic (i.e. towards the Internet) is handled reasonably well in a static scenario. However, it uses the same upward route to handle downlink communications also. This is not at all an efficient procedure, considering the dynamic nature of LLNs. Increase in overhead and delay associated with LOADng and opportunistic routing protocols also need due consideration.

- **Distributed Control** – Any networking procedure, not only routing, can follow either a centralized or a distributed mode of operation. In centralized approach, a single entity will be responsible for taking decision, be it a routing decision or a decision regarding security. Distributed approach, on the other hand, relies on individual nodes to take decisions. As far as LLNs are concerned, a shift from centralized to a distributed approach seems mandatory.

  The disadvantages of a centralized approach can be best explained using the context of non-storing mode of operation in RPL. The fact that root node alone stores the routing information, results in severe congestion surrounding the root. Considering the resource constrained nature of LLNs, such a congestion can result in the rapid depletion of node batteries in that region. Hence, suitable measures need to be taken to avoid such a condition.

- **Multicast Routing** – Multicast is an important form of communication in networks, along with unicast, broadcast and flood operations. It involves a sender transmitting messages to more than one recipients. Like in any other kind of a network, multicast is essential in LLNs also, for realizing a variety of IoT applications.

  There are considerable IoT applications which can benefit from efficient multicast operations. Smart street lighting mechanism can provide a convincing example. Such a system requires a sensor to send signals to an array of streetlights, so as to turn their actuators on simultaneously. This is equivalent to a classical multicast scenario where the sensor acts as the source, and actuators in multiple streetlights act as destinations. The current trend of replacing a multicast operation with multiple unicast operations is highly inefficient. Taking into consideration these issues, the IETF has drafted a *Multicast Protocol for Low Power Lossy Networks (MPL)* in 2016 [32]. Still, the development of MPL is in initial stages, and requires more enhancements.

- **Quality of Service (QoS)** – QoS is often used to measure the level of performance obtained from an application. With the IoT applications having a largely societal outlook, ensuring the desired level of QoS is essential. Being an enabling technology of the IoT, LLNs need to be concerned about primarily two variants of QoS:
  - **Multi-dimensional QoS** – Due to the presence of applications with different QoS requirements in the same network, there could be more than one root nodes. Each root node will be assigned the task of handling a specific type of application (and thereby a specific QoS). As a result, multiple source-destination pair will have different QoS requirements. Hence, a routing protocol designed for an LLN should operate accordingly.
  - **Multi-traffic QoS** – It is quite impossible to have a scenario where each and every node in an LLN generate the same kind of traffic. Instead, the types of traffic in a LLN can vary from bursty traffic generated by smart meters to continuous traffic generated through video streaming. An LLN routing protocol should be able to handle the unique characteristics pertaining to each node.

- **Multiple Routing Instances** – We have already seen that multiple applications can coexist in the same LLN. A DODAG will be constructed by each of these applications, as per the QoS requirements. Thus, an LLN can be visualized as consisting of a number of DODAGs, both overlapping as well as non-overlapping. Such a condition, if not addressed properly, can often lead to poor and inefficient utilization of resources.

  As already mentioned, DODAGs can either be overlapping or non-overlapping. In more technical terms, such a scenario offers multiple virtual networks sharing the same physical devices. If the routing procedure is not designed to consider this, nodes present in multiple DODAGs may drain out their resources quickly, and thereby result in the reduction of network lifetime.

- **Security** – With everyday 'things' getting connected to the Internet, there is an inherent risk of exposing the private information of users into the public world. Obviously, there is a need to incorporate cryptographic operations in the routing protocol. But, the computationally intensive nature of cryptographic operations is not suited for the resource constrained nodes in LLNs. Hence, lightweight security operations need to be incorporated in LLN routing protocols.

The above challenges need to be given due consideration, especially with an 'IoT rush' already been predicted by researchers. With billions of 'things' getting connected to the Internet by 2020, development of scalable and efficient LLN routing protocols is surely a matter of concern.

6. Conclusion

An 'IoT rush' by the year 2020 has already been foreseen by researchers. With more number of 'things' getting attached to the Internet of Things (IoT), there will be an obvious increase in the number of applications demanding the service of such a paradigm. Low Power Lossy Networks (LLNs), being an enabling technology of the IoT, play a significant role in the realization of such a huge range of applications. Considering the resource constrained nature of nodes present in LLNs, the routing protocols should ensure a high level of resource efficiency, or else it can adversely affect the timely performance of IoT applications.

This paper presents a detailed review on various LLN routing protocols like Routing Protocol for LLNs (RPL), LLN On-demand Ad hoc Distance vector routing protocol – next generation (LOADng), and opportunistic routing. The review covers recent as well as important enhancements made to the core protocol, be it RPL, LOADng, or opportunistic routing. These protocols have been analyzed with respect to the degree of resource management achieved by them. In addition to that, important LLN routing metrics have been studied, and a mapping between metrics and resources managed has been provided. Finally, the important challenges that signal an immediate attention of researchers have been enlisted. Considering the rapidly developing nature of IoT paradigm, there is an ample scope as well as need for the development of efficient routing protocols.

References

[1] M. Richardson and I. Robles, "RPL – routing over low power and lossy networks", IETF 94.

[2] T. Winter et al., "RPL: IPv6 routing protocol for low-power and lossy networks," IETF RFC 6550, Mar. 2012.

[3] T. Clausen et al., "The lightweight on-demand ad hoc distance-vector routing protocol – next generation (LOADng)", Internet Draft, work in progress, draft-clausen-lln-loadng), 2013.

[4] M. Zhao, A. Kumar, P. Chong, and R. Lu, "A comprehensive study of RPL and P2P-RPL routing protcols: implementation, challenges, and opportunities", Peer-to-Peer Netw. Appl., Springer, July 2016, pp. 1232 – 1256.

[5] A. Oliveira and T. Vazao, "Low-power and lossy networks under mobility: a survey", Computer Networks, Elsevier, April 2016, pp. 339 – 352.

[6] D. Airehrour and J. Gutierrez, S. K. Ray, "Secure routing for internet of things: a survey", Journal of Network and Computer Applications", March 2016, pp. 198 – 213.

[7] J. Guo, P. Orlik, K. Parsons, K. Ishibashi, and D. Takita, "Resource aware routing protocol in heterogeneous wireless machine-to-machine networks", IEEE Global Communications Conference (GLOBE-COM), 2015, pp. 1 – 6.

[8] J. Ko, J. Jeong, J. Park, J. A. Jun, O. Gnawali, and J. Paek, "Dual-MOP-RPL: supporting multiple modes of downward routing in a single RPL network", ACM Transactions on Sensor Networks (TOSN), vol. 11, no. 2, March 2015, pp. 39:1 – 39:20.

[9] C. Kiraly, T. Istomin, O. Iova, and G. P. Picco, "D-RPL: overcoming memory limitations in RPL point-to-multipoint routing", IEEE 40th Conference on Local Computer Networks (LCN), October 2015, pp. 157 – 160.

[10] K. Vyas, J. Sengupta, and S. D. Bit, "ARPL: supporting adaptive mixing of RPL modes to overcome memory overflow", IEEE International Symposium on Smart Electronic Systems (iSES), 2018, pp. 128 – 133.

[11] K. C. Lee, R. Sudhaakar, L. Dai, S. Addepalli, and M. Gerla, "RPL under mobility", IEEE consumer communications and networking conference (CCNC), 2012, pp. 300 – 304.

[12] C. Cobrzan, J. Montavont, and T. Nol, "Analysis and performance evaluation of RPL under mobility", IEEE symposium on computers and communications (ISCC), 2014, pp. 1 – 6.

[13] H. Fotouhi, D. Moreira, and M. Alves, "MRPL: boosting mobility in the internet of things", Ad Hoc Networks, Elseveir, 2015, pp. 17 – 35.

[14] H. Fotouhi, D. Moreira, M. Alves, and P. M. Yomsi, "MRPL+: a mobility management framework in RPL/ 6LoWPAN", Computer Communications, Elsevier, 2017, pp. 34 – 54.

[15] S. Sanshi and C. D. Jaidhar, "Enhanced mobility aware routing protocol for low power and lossy networks", Wireless Networks, Springer, 2017.

[16] Y. Tahir, S. Yang, and J. McCann, "BRPL: backpressure RPL for high-throughput and mobile IoTs", IEEE Transactions on Mobile Computing, 2018, pp. 29 – 43.

[17] H. Kharrufa, H. Al-Kashoash, and A. H. Kemp, "A game theoretic optimization of RPL for mobile Internet of Things applications", IEEE Sensors Journal, Vol. 18, No. 6, 2018, pp. 2520 – 2530.

[18] S. Sanshi and C. D. Jaidhar, "Enhanced mobility routing protocol for wireless sensor network", Wireless Networks, Springer, 2018.

[19] M. Goyal, E. Baccelli, M. Philipp, A. Brandt, and J. Martocci, "Reactive discovery of point-to-point routes in low power and lossy networks (draft-ietf-roll-p2p-rpl-09)". Technical Report, Internet Draft, Work in Progress, 2012.

[20] M. Goyal, A. Brandt, and E. Baccelli, "A mechanism to measure the routing metric along a point-to-point route in a low power and lossy network", RFC 6998, 2013.

[21] M. Goyal, M. Philipp, A. Brandt, and E. Baccelli, "Reactive discovery of point-to-point routes in low power and lossy networks", RFC 6997, 2013.

[22] T. Clausen, J. Yi, and U. Herberg, "Lightweight on-demand ad hoc distance-vector routing – next generation (LOADng): protocol, extension, and applicability", Computer Networks, Elsevier, 2017, pp. 125 – 140.

[23] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking: data forwarding in disconnected mobile ad hoc networks", IEEE Communications Magazine, 200, pp. 134 – 141.

[24] S. K. Dhurandher, S. J. Borah, I. Woungang, A. Bansal, and A. Gupta, "A location prediction-based routing scheme for opportunistic networks in an IoT scenario", Journal of Parallel and Distributed Computing, Elsevier, 2017, http://dx.doi.org/10.1016/j.jpdc.2017.08.008.

[25] www.cse.chalmers.se/~olafl/papers/2013-02-ewsn-duquennoy-orpl.pdf, accessed on 29th August 2018, 7: 08 PM IST.

[26] M. Michel, S. Duquennoy, B. Quoitin, and T. Voigt, "Load-balanced data collection through opportunistic routing", 2015 International Conference on Distributed Computing in Sensor Systems, IEEE, 2015.

[27] S. Gormus, Z. Fan, Z. Bocus, and P. Kulkarni, "Opportunistic communications to improve reliability of AMI mesh networks", 2nd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies, IEEE, 2011.

[28] S. Gormus, F. Tosato, Z. Fan, Z. Bocus, and P. Kulkarni, "Opportunistic RPL for reliable AMI mesh networks", Wireless Networks, Springer, 2014.

[29] P. Karkazis et al., "Design of primary and composite routing metrics for RPL-compliant wireless sensor networks", International Conference on Telecommunications and Multimedia (TEMU), 2012.

[30] R. Baumann, S. Heimlicher, M. Strasser, and A. Weibel, "A survey on routing metrics", TIK Report 262, ETH-Zentrum, Computer Engineering and Networks Laboratory, 2007.

[31] K. S. Hong and L. Choi, "Dag-based multipath routing for mobile sensor networks", ICTC, 2011, pp. 261 – 266.

[32] J. Hui and R. Kelsey, "Multicast protocol for low-power and lossy networks (MPL)", IETF RFC 7731, February 2016.

Dr. S. D Madhu Kumar is a Professor in the Department of Computer Science & Engineering of National Institute of Technology Calicut. For the past 27 years he has been working in the Computer Science & Engineering Department of National Institute of Technology Calicut (NIT Calicut), which is a centrally funded institute of National Importance in India. He was the Head of the Computer Science & Engineering department of NIT Calicut during 2012-2014. Currently he is the Dean (Students' Welfare) of NIT Calicut. Dr. S.D Madhu Kumar received his PhD in Computer Science & Engineering from the Indian Institute of Technology Bombay (IIT Bombay). Earlier he did his Masters in Engineering in Computer Science & Engineering from Indian Institute of Science, Bangalore (IISc Bangalore), India. He has more than 60 International publications to his credit and has produced seven PhDs and currently guiding six Doctoral students at NIT Calicut. His research interests include Cloud Computing, Big Data Analytics, Software Engineering and DBMS. He is a senior member of IEEE and a member of ACM USA. He has been the chair and keynote speaker for many International Conferences. He has been the Track Co-chair of the Cloud Computing Track of the ACM Symposium on Applied Computing (SAC) since 2010. He also serves in the editorial board of several international journals including ACM ACR. More details of Dr. Madhu can be found at http://people.cse.nitc.ac.in/madhu

Jaisooraj J is currently pursuing his PhD in the Department of Computer Science and Engineering, National Institute of Technology Calicut, India. Earlier, he received his B. Tech in Computer Science and Engineering from Kerala University, India in the year 2012, and M. Tech in Computer Science and Engineering from Kerala University, India, in the year 2015. His research interests include protocols for computer networks, wireless networks, and Internet of Things.

Aindrila Ghosh is currently pursuing Masters in Computer Science and Engineering at National Institute of Technology Calicut, India. She secured her B. Tech in Information Technology from West Bengal University of Technology, India in the year 2014. Her research interests include computer networks, and protocols for Internet of Things.