A review on routing protocol for low power and lossy networks in IoT

Amol Dhumane
Dept. of Computer Engineering,
NBN Sinhgad School of Engg.
Pune, India
amol.dhumane@sinhgad.edu

Avinash Bagul
Dept. of Computer Engineering,
NBN Sinhgad School of Engg,
Pune, India
avinash.bagul@sinhgad.edu

Parag Kulkarni
Dept. of Computer Engineering,
NBN Sinhgad School of Engg,
Pune, India
parag.kulkarni@sinhgad.edu

***Abstract*: In Low-Power and Lossy Networks (LLNs) all the devices including the routing ones are constrained in terms of processing power, memory and the energy. Due to the lossy nature of the networks, losses may occur while transferring the data. These networks are facing the problems in terms of data transfer rates due to the constraints in the devices. In this paper we are going to analyze the working of Routing Protocol over Low Power and Lossy Network (RPL), which is considered as a de-facto routing standard in the Internet of Things. RPL is a distance vector routing algorithm that uses IPV6 as a routing protocol; it supports point to point, point to multipoint and multipoint to point traffic patterns.**

***Keywords*: Routing protocol for low power and lossy networks (RPL), Destination oriented directed acyclic graph (DODAG), Internet of Things (IoT)**

I. INTRODUCTION

RPL is designed for LLN networks which are having tremendous applications in various domains. LLN's power consumption is more because they need to do retransmissions of the packets due to collisions and packet loss due to its lossy nature. RPL separates the packet processing and forwarding from the routing optimization objectives such as minimizing the energy consumption, minimizing the communication delays or satisfying the constraints [1]. Several studies such as [3] and [5] reveal the impact of wireless lossy links on the overall reliability, power efficiency and maximum achievable throughput. RPL protocol is able to quickly build network routes, to distribute routing knowledge among nodes. RPL is designed by the Routing over Low Power and Lossy (ROLL) Networks group of International Engineering Task Force (IETF) for resolving the routing issues in low power and lossy networks. It implements various measures for reducing the overall energy consumption by the dynamic sending rate of control messages. It also addresses topology inconsistencies. Multiple instances of RPL may run concurrently in the network. Each such instance may provide different performance criteria.

In the typical network settings, the collected data is sent to the root or sink node by the proper coordination among the routing nodes through multihop paths. There can be a small set of sink stations. DADOG's are formed for data routing to the base stations. Normally the sink stations are the devices having high processing power, computing abilities and without energy constraints. The data analysis and knowledge generation is done at the sink stations. The IP for Smart Objects (IPSO) Alliance has done a great amount of effort to endorse the use of IP for small and constrained devices. This is the top organization which defined the Internet of Things and it supports the use of the layered IP architecture for small computers. Along with IETF IPSO is trying to for the adoption of IPv6 over LLN's.

The rest of this paper is organized as follows. Section 2 describes RPL's terminology and the basic features. Section 3 gives an idea about topology construction, control message and its structure is discussed in section 4. Section 5 briefs about routing metrics and constraints.

II. PROTOCOL OVERVIEW

LLN's do not have predefined topologies. RPL arranges topology as a Destination Oriented Directed Acyclic Graph (DODAG). Each sink node is having one DODAG associated with it. The sink node is also called as DODAG root. As RPL forms DODAG's, it means that network devices running this protocol are connected in such a way that cycles are absent in the routing paths. RPL protocol supports a loop detection and avoidance mechanism.

For maintaining and identifying a topology RPL uses four key values:

1. RPLInstanceID: RPLInstanceID is a set of one or more DODAG's. For each RPLInstanceID one objective function is defined. A network may contain one or more RPLInstanceIDs. The set of DODAGs identified by a RPLInstanceID is called a RPL Instance.
2. DODAGID: As RPLInstance has multiple DODAGs associated with it. The combination of RPLInstanceID and DODAGID uniquely identifies a DODAG.
3. DODAGVersionNumber: In the process of network formation or while maintaining the network topology, sometimes the DODAGs are reconstructed from the DODAG root which results into increase in its version number. DODAG version is uniquely identified with the help of RPLInstanceID, DODAGID and DODAGVersionNumber.

Rank: It defines the individual nodes position with respect to DODAG root. The correct estimation of the Rank is the responsibility of the Objective Function. It is used for avoidance and detection of the loops.

Figure 1 shows these four key values and the relation between them.

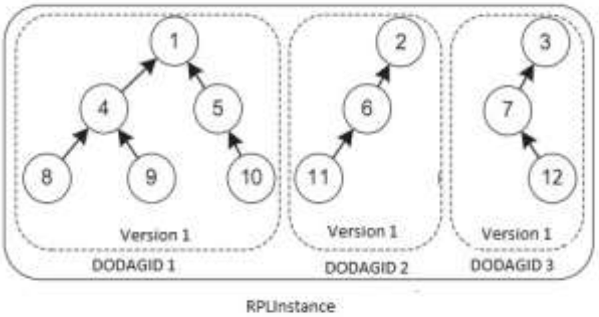

Fig.1: RPL Instance

Even though the figure shows a simple tree like structure, but in practical scenario DODAG allows each node to have multiple parents.

Objective functions are used while selecting and optimizing the routes based on various constraints. It defines how RPL nodes select and optimize routes within a RPL Instance [1]. It calculates the ranks of the nodes based on the metrics and constraints defined in it.

The protocol contains global and local topology repair mechanisms. Global repair mechanism starts from the DODAG root and which may result into increase the DODAG version number. Global repair has the cost of additional control traffic in the network. In this case each node will re-execute the objective function for selecting the preferred parent. The local repair mechanism takes place within the same DODAG version. In this case the graph may start to diverge from its optimum shape.

RPL deals with infrequent data traffic due to that maintaining routing topology constantly up to date will result into energy wastage. So RPL does not address the temporary changes in the topology until there is data to send.

RPL contains three security modes: unsecured, preinstalled and authenticated. Unsecured mode uses the already present security mechanisms such as link layer security to meet the security requirements. In preinstalled mode, nodes are having preinstalled keys. These keys help the nodes to generate secure RPL messages while joining RPLInstance as a leaf node. In authenticated mode, joining an authenticated RPL Instance as a router requires obtaining a key from an authentication authority.

The DODAGs can be grounded or floating. The grounded DODAG provides connectivity for satisfying application-defined goals where as the floating DODAGs preserves connectivity during the repair mode.

RPL topologies contain mainly three types of nodes: root, routers and leaf nodes. Root nodes provide connectivity to the other networks. Routers provide the routing information to its neighbors and leaf nodes listen for DIOs and use their information for joining the DODAG.

III. TOPOLOGY FORMATION AND UPWARD ROUTING

Following are the common rules used during the topology construction:

a. The parent set of DODAG root is always empty.
b. The DODAG parent set of any node must be a subset of its neighbor set.
c. The node’s preferred parent must be a member of its DODAG parent set.
d. The Ranks of the parent nodes of a given node must be smaller than that of the Rank of the node.
e. The routes going through the unreachable neighboring node must not be considered in the routing process. Such routes must be removed from the routing table.

RPL uses following four main types of messages for the topology formation and its maintenance:

1. DODAG Information Object (DIO): Used in case of path creations for upward routing i.e. multipoint to point.
2. Destination Advertisement Object (DAO): Used in case of path creations for downward routing i.e. point to multipoint. It is used to propagate destination information Upward along the DODAG.
3. DODAG Information Solicitation (DIS): Used to solicit or request a DIO from a RPL node. A node may use DIS to search its neighborhood for nearby DODAGs.
4. Destination Advertisement Object Acknowledgment (DAO-ACK): It is sent as a unicast packet by a DAO recipient (a DAO parent or DODAG root) in response to a unicast DAO message.

Topology formation process starts at the root node which is also called as border gateway by sending the DIO messages. The neighboring nodes of root receive and process the messages. A decision regarding to joining the topology is made. Appropriate parent selection is done based on the metric and the constraints defined by the objective function. Once the appropriate parent is selected, a route gets established between the current node and the DODAG graph root. If a node is router, it computes its own rank and updates its rank in the DIO messages which it is going to transmit to its neighbors. The Rank of a node is a scalar representation of the location of that node within a DODAG Version [1].

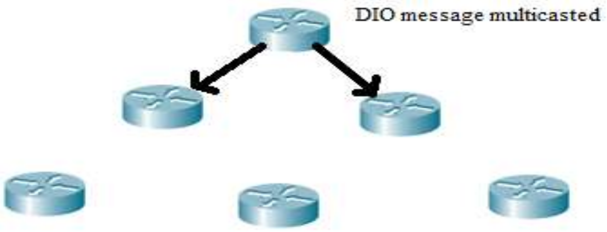

Fig 2(a): step I

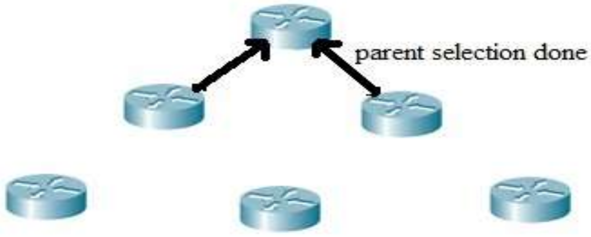

Fig 2(b): step II

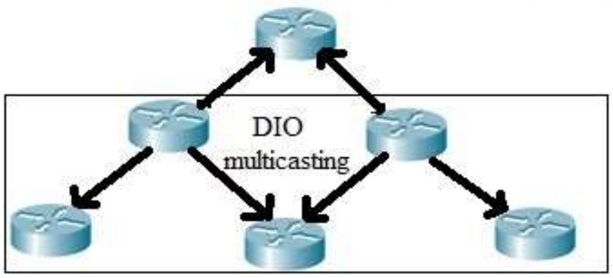

Fig 2(c): step III

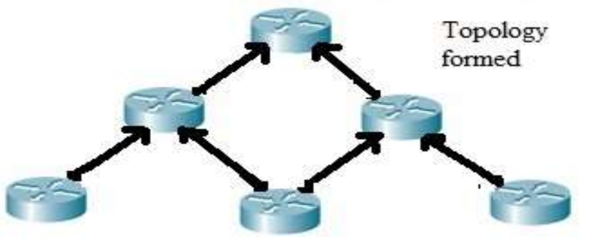

Fig 2(d): step IV

The process of transmitting the DIO message and parent selection continues till it hits the leaf nodes. This rippling effect builds the DODAG topology from root node to the leaf nodes.This use of DIO messages is mainly for establishing the paths for upward routing. For creating downward paths, another mechanism is essential which makes use of DAO messages. In the DODAG formation process each node of the graph has a routing entry towards its parent or multiple parents depending on the objective function in a hop-by-hop fashion and the leaf nodes can send a data packet all the way to root of the graph by just forwarding the packet to its immediate parent [4]. RPL supports constraint-based routing where constraints may be applied to both link and nodes [1]. It includes the loop avoidance mechanism during the topology changes by using rank based data path validation mechanisms.

Rank of the node can be computed by the equation 1 where MinHopRankIncrease is the minimum increase in Rank between a node and any of its DODAG parents.

$$DAGRank(rank) = floor\left(\frac{rank}{MinHopRankIncrease}\right) \quad (1)$$

A very large MinHopRankIncrease, for example, allows accurate characterization of a given hop's effect on Rank but cannot support many hops [1]. When Rank is compared for the determination of parent relationships or in loop detection process, only the integer portion of the Rank is used. Less the rank of the node more it’s closer to the DODAG root node.

RPL has the ability of multi-topology routing (MTR) with the help of DODAG instance-id. The idea is to construct and identify multiple graphs (DODAGs) over the same physical topology [4]. A node can only join a single graph within an RPL instance-id but can be associated with several RPL instance-ids simultaneously [4].

Some nodes can show greediness in their behavior. They may move deeper in the DODAG for increasing their Rank as well as for increasing the size of their parent set. Because of the greediness the Ranks of the node may increase till infinity and it may result into instability of the DODAG. So once a node is joined a DODAG Version, RPL disallows greediness, for preventing the resulting instabilities in the DODAG Version [1].

There can be multiple and logically independent RPL instances in the LLN. A RPL node may be a part of more than one RPL Instance. The node can act as a router in some of the instances and it can act as a leaf node in the other instances. There are two types of RPL Instances: Local and Global. Global RPL Instances are coordinated with one or more DODAGs and are mostly long-lived. Local RPL Instances have always a single DODAG associated with them. Data and control packets are tagged properly for uniquely identifying the RPL Instance.

IV. RPL CONTROL MESSAGES

Figure 3 shows a RPL control message which is a new ICMPv6 message. The Type field specifies whether RPLInstanceID is Global or Local. Code field identifies type of RPL control message. Out of total nine types four are already mentioned in section III. Remaining five types can be: Secure DODAG Information Solicitation, Secure DODAG Information Object, Secure Destination Advertisement Object and Secure Destination Advertisement Object Acknowledgment.

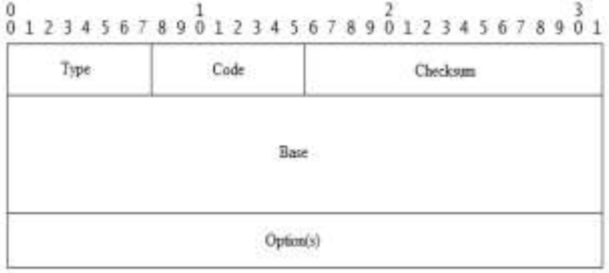

Fig 3: RPL Control Message

If a node receives a RPL control message with an unknown Code field, the node must discard the message without any further processing or may raise a management alert and must not send any messages in response [1].

The Base field is the RPL message header and it contains the basic information related to the functions of the carried object [2]. The Options field is the body of such messages. Figure 4 shows a format of secure RPL messages to support confidentiality and integrity.

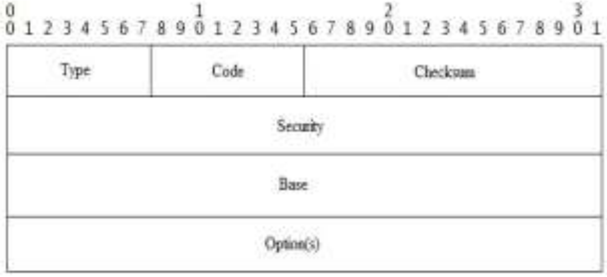

Fig 4: Secure RPL Control Message

V. ROUTING METRIC, CONSTRAINTS AND OBJECTIVE FUNCTION

Metric and constraints can be node (such as energy of the node, hop count etc) as well as link (such as latency, reliability, throughput etc) based. According to the type of the data traffic a sophisticated metric is required for the LLNs for routing purpose. Metric is used for choosing the best possible path. On the other hand a constraint is used to cut the links or nodes from the DODAG which do not meet the set of constraints. Routing protocol reacts according to the changes in metric and constraints. On the other hand, it is essential to control the alteration rate of routing metrics in order to avoid path instabilities, which can severely harm LLN performance.

The Objective Function is indicated in the DIO message using an Objective Code Point (OCP), and it indicates the method that must be used to construct the DODAG [1]. Objective functions translates the key metrics and constraints into Rank, which models the node distance from a DODAG root, in order to optimize the network topology in a very flexible way [2]. Objective function does two main tasks:

a. It does selection of a DODAG to join and
b. It helps to recognize the number of peers in that DODAG as parents.

Objective function selects parents for a node when one of the following events is triggered:

a. Reception of DIO message.
b. A timer elapses
c. All the parents of the current node in the DODAG are currently unavailable.
d. State of the candidate neighbor is changed.

After selecting the parent set, the Rank of every parent is evaluated. The parent node which is having lowest Rank is selected as a preferred parent. While choosing the preferred parent it is necessary to make sure that the Rank of its parent node must be less than that of the current node.

VI. LOOP DETECTION AND AVOIDANCE

Loop detection and avoidance mechanism in RPL makes it special compared to the traditional networks. Loops may form for a number of reasons such as loss of control packets. It is necessary to detect the loops as early as possible for avoiding packet drops caused by the TTL expiry. RPL loop avoidance mechanisms are kept simple. RPL contains a reactive loop detection technique. LLNs have low data rate compared to the traditional networks. So the impact of the temporary loops will be limited on LLNs. In such situations it is advisable to under-react the situation, as the loops could be temporary. Over-reacting may result into routing oscillations and energy consumption in nodes due to excess number of control packets used for avoiding loop. RPL never guarantees loop free topology due to that it provides the loop avoidance and detection mechanism.

Routers multicast DIO messages for topology setup as well as maintenance. The nodes in the range of the router accept these DIO messages for computing its parent nodes set. In this process there is always a danger to a node of selecting its own child as a parent. Therefore RPL node does not process the DIO messages coming from those nodes whose Ranks are higher than that of it.

Loops in LLNs are unavoidable hence there is a need for detecting these loops in addition to loop avoidance rules [4]. This can be achieved by setting the bits such e.g. 'down' bit in RPL header and processing these bits as a part of data-path validation.

VII. TRICKLE TIMER

Traditional routing protocols update their routing table periodically. This periodic update mechanism is not useful in RPL as LLNs are resource constraint networks. RPL uses trickle timer mechanism [6] which is adaptive in nature. It controls the sending rate of DIO messages which are responsible for topology formation or its maintenance. The algorithm treats building of graphs as a consistency problem and makes use of trickle timers to decide when to multicast DIO messages [4]. When the network gets stable the interval of the trickle time increases and whenever the inconsistencies are increased the interval decreases. Higher value of trickle timer results into less transmission of DIO control messages and less value of it produces more DIO control messages.

VIII. DOWNWARD ROUTING

RPL uses Destination Advertisement Object (DAO) messages for setting up Downward routes. DAO messages are an optional feature for applications that require point-to-multipoint (P2MP) or point-to-point

(P2P) traffic [1]. For setting up Downward routes RPL nodes send the DAO messages Upward to its DAO parent. The structure of the DAO message is shown in figure 5.

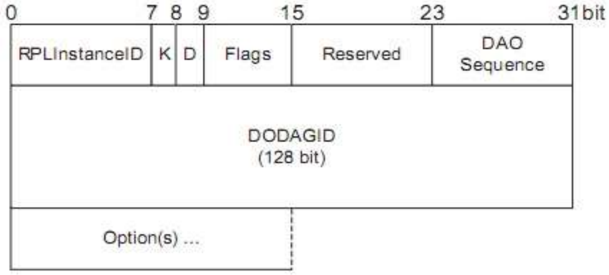

Fig 5: DAO message structure

In DAO, message contains RPLInstance ID similar to DIO message. The K flag is used to inform the receiver to reply back to the sender by DAO-ACK. The D flag indicates the DODAGID field is present. The 6 bit Flag field and 8 bit Reserved fields are generally initialized to zero by the senders and are ignored by the receivers. DAOSequence field is incremented at each unique DAO message from a node and echoed in the DAO-ACK message. DODAGID field is present when the 'D' flag is set. This field is present when a local RPLInstanceID is in use for identifying the DODAGID which is associated with the RPLInstanceID.

RPL supports two modes of Downward traffic: Storing (fully stateful) or Non-Storing (fully source routed). In storing mode each node stores Downward routing tables for their sub-DODAG. Each node on a Downward route in a storing network observes its routing table to choose the next node for sending the data packet. In non-storing mode the nodes do not store Downward routing tables. In this mode each node has to propagate the list of its parent to the root node. The root node after receiving this topology information computes the paths to the destinations.

IX. CONCLUSION

The drastic change is occurred in our lives due to the internet. The internet is constantly changing and we will be witness of it in the coming decade or two. Sizes of the devices are getting smaller and smaller and the number of devices are connected to the internet are getting increased with a dramatic rate. The future is IoT. We need a good routing solution for routing the huge data that is going to be produced by these huge numbers of devices which is supporting IPv6 addressing mechanism. RPL provides a solution on it. RPL is developed to support and fulfill these requirements. RPL runs on the nodes having limited energy and processing capabilities. It tries to save the energy of the nodes by sending less number of control frames. It sends the control frames on the network adaptively instead of sending it at regular time intervals as that of the traditional routing protocols. It supports various types of traffic patterns such as P2MP, MP2P and P2P. Overall, it can provide a good solution on routing in the coming IoT era.

ACKNOWLEDGEMENT

There have been many contributors in this work and the authors are thankful to each of them. We specifically would like to thank Dr. S.D. Markande, Dr. J.R. Prasad and Dr. P.N. Mahalle for their constant support and motivation in our work.

REFERENCES

[1] T. Winter et al., “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks”, RFC 6550, IETF, 2012.

[2] Maria Rita Palattella, Nicola Accettura, Xavier Vilajosana, Thomas Watteyne, Alfredo Grieco, Gennaro Boggia, Mischa Dohler, “Standardized Protocol Stack for the Internet of (Important) Things”, IEEE Communications Surveys and Tutorials, IEEE, PP(99):1–18, 2012. ISSN 1553-877X. doi:10.1109/SURV.2012.111412.00158.

[3] A. Cerpa, J. Wong, L. Kuang, M. Potkonjak, and D. Estrin, “Statistical model of lossy links in wireless sensor networks”, Information Processing in Sensor Networks, IPSN 2005, pages 81–88, 2005.

[4] J. Vasseur, N. Agarwal, J. Hui, Z. Shelby, P. Bertrand, and C. Chauvenet, “RPL: The IP routing protocol designed for low power and lossy networks,” Internet Protocol for Smart Object (IPSO) Alliance, White Paper, April 2011.

[5] Y. Li, J. Harms, and R. Holte, “Impact of lossy links on performance of multihop wireless networks”, Computer Communications and Networks, 2005. ICCCN 2005, pages 303–308, 2005.

[6] P. Levis, N. Patel, D. Culler, and S. Shenker, “Trickle: A self-regulating algorithm for code maintenance and propagation in wireless sensor networks”, USENIX NSDI Conference, pages 15–28, San Francisco, CA, USA, 2004