IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol-6, Special Issue-4, May 2018 E-ISSN: 2347-2693

QoS Measurement of RPL using Cooja Simulator and Wireshark Network Analyser

A.S. Joseph Charles[1], P. Kalavathi[2*]

[1] Department of CSA, The Gandhigram Rural Institute – Deemed to be University, Gandhigram, India
[2*] Department of CSA, The Gandhigram Rural Institute – Deemed to be University, Gandhigram, India

**Corresponding Author: pkalavathi.gri@gmail.com, Tel.: +91-94458-98371*

Available online at: www.ijcseonline.org

Abstract— The Internet of Things (IoT), with its ability to collect data using sensors and store the voluminous data over the cloud has become the de facto standard in building up smart homes and smart cities. The routing protocols are used in the network layer and they play the pivotal role. They perform the intelligent task of forwarding and routing. If the routing is not done properly then there will be a heavy loss and retransmission of the packets, that would cost more power, memory, bandwidth and procession capacity. Therefore, the routing protocols used in the regular networks cannot be used efficiently in IoT. IPv6 routing protocol for Low power and lossy networks (RPL) has become the favourite routing protocol of Internet of Things. There are several metrics used in the RPL to determine the path cost and to help to connect the nodes with each other. The performance quality of RPL can be analysed and measured from the factor that how best it works utilizing the resources like energy, memory, bandwidth etc. The quality of services parameters like packet delivery ratio, network convergence time, remaining energy, latency and control traffic overhead are analysed to measure the performance of RPL. The Cooja simulator running over the Contiki Sensor OS is chosen as an ideal platform due to its special feature of supporting the cross-level simulation. The open source network analyser Wireshark used in Contiki OS also helps in the process of performing the protocol analysis.

Keywords— Internet of things, Routing, Low power and lossy networks, RPL, QoS for RPL, Cooja Simulator, Wireshark

I. INTRODUCTION

The emergence of Internet of Things (IoT) Technology is going to create a great revolution in the field of networking. In IoT a large number of devices, objects and computers are interconnected using various connecting technologies like Zigbee, Bluetooth, Wi-Fi and GSM. These different ways and possibilities of connecting are provided in the link layer of the IoT with IEEE 802.15.4, which is the standard for link layer frames delivery in the low power and lossy networks [1]. In the conventional networks, we usually have connectivity between homogeneous devices, but in IoT, there is connectivity between heterogeneous devices and networks [2]. The wireless sensor networks, when enabled with the ability to store data on the cloud, it is called Internet of Things. In IoT, sensors are used in some way or the other to collect data. The devices in IoT are called nodes and they use minimum energy and usually run for years on small and inexpensive batteries [3].

There is a stack of protocols used in the network to control the data. Among the layers of network, the network layer is the one which controls the flow of data between the nodes [4]. The protocols used in the regular networks cannot be used in the constrained network of IoT, due to the scarcity of memory, energy and processing capabilities available in IoT devices. The protocols such as Open Shortest Path First Protocol (OSPF) Protocol, Intermediate System to Intermediate System (IS-IS) Protocol, Optimized Link State Routing (OLSR) Protocol and Ad hoc On demand Distance Vector protocol (AODV) did not satisfy the needs of the low power and lossy networks, though they were efficient in the regular and conventional networks [5]. The routing protocols do the job of forwarding the packets and routing. If these jobs are not performed intelligently then there will be loss of packets and retransmission of the packets, costing more memory, bandwidth and procession power. Therefore, the protocols of the normal network cannot be used here and we adopt RPL to do this job [6].

RPL has become the de facto routing protocol in IoT, due to its advantages over the other routing protocols. RPL has a better response time, because the routes are readily available. RPL has only the local routing information, therefore it does not flood the network, and it is scalable. RPL can be used in a non-infrastructure network [07].

A. RPL – Routing Protocol for LLNs

Low power and Lossy Networks (LLN) consist of constrained nodes that have limited energy, memory and processing capacity. RPL is designed in such a way that multiple RPL instances can run at a time and the packet forwarding is separated from routing optimization, in order to support LLN. Topology construction is one of the key objectives of RPL, because LLN do not usually have predefined topologies. RPL fixes one or more roots to function as a sink and then forms routes from or towards the sinks. The resultant routes form a Directed Acyclic Graph (DAG) as a topology, that is further partitioned into few Destination Oriented Directed Acyclic Graphs (DODAGs). There will be only one DODAG per sink [08].

1) RPL Identifier

There are four values used to identify and maintain the topology in RPL.

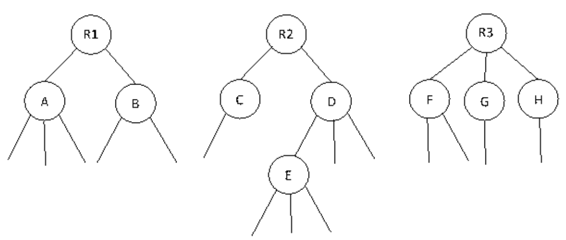

Figure 1. RPL Instance

1. RPLInstanceID: This ID identifies the set of DODAGs. A network may have multiple RPLInstanceIDs, one for each objective function. We name the set of DODAGs identified by an objective function as RPL Instance [08].

2. DODAGID: This ID is used to uniquely identify a DODAG in the network.

3. DODAG Version Number: DODAG is reconstructed from the root, by increasing this version number.

4. Rank: This is a number which defines the distance of a node from the DODAG root.

A RPL instance in the network may be i) a single rooted DODAG ii) Multiple rooted DODAG iii) A single DODAG with virtual root iv) A combination of the above three. Fig. 1 shows the different types of RPL Instance. The DODAG with one type may switch over to other types in the course of time. Especially in mobile sensor networks the nodes are constantly moving and therefore the topology would also be changing frequently, creating more control traffic overhead.

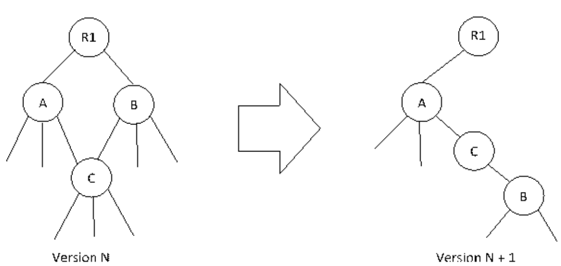

Figure 2. DODAG Version

Fig. 2 depicts the conversion of the DODAG from one version to another. Control messages are used in constructing and maintaining the topology. The general format of RPL control message is shown in Fig. 3.

2) Control Messages

There are four types of control messages:

1) DODAG Information Solicitation (DIS) – It is used to look for a DIO from the RPL node.

2) DODAG Information Object (DIO) – It is the carrier of information regarding the RPL instance and its configurations.

3) Destination Advertisement Object (DAO) – It is used to propagate the information regarding destination to the upward nodes.

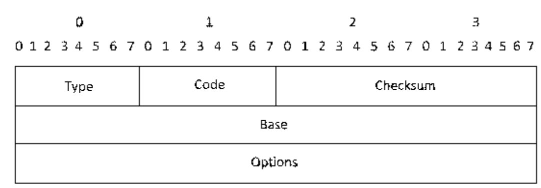

Figure 3. RPL Control Message

4) Destination Advertisement Object Acknowledgement (DAO-ACK) – It is used in unicast communication in response to a unicast message.

3) Objective Function

The objective function defines, guides and directs the RPL nodes in constructing and optimizing the routes within a RPL instance. The objective function also defines how each node should translate the specified metrics and constraints in forming the routes [08]. The routing metrics are the quantitative values used to measure the path cast. The metrics may be link metric or the node metric. Link metrics are used to measure the quality of the links existing between the nodes, whereas the node metrics are the quantitative values of the node properties. These metrics are usually additive. Some metrics may also be qualitative and dynamic or static. The values also can be used as metrics, as it is, or as constraints, conforming to a threshold value. The metrics used in the RPL are 1) Node state and attribute object ii) Node energy object iii) Hop count iv) Throughput v) Latency vi) ETX (Expected Transmission Count) vii) Link color [09]. RPL supports two objective functions based on the metrics hop count and ETX. Zero objective function (OF0) uses the metric, hop count and Minimum Rank with Hysteresis Objective Function (MRHOF) uses the Expected Transmission Count (ETX).

4) Rank Calculation

Some metrics are used to assign rank and choose the preferred parent based on the rank. Each node moves between being a node and parent depending on the Rank it holds. Rank is a 16-bit integer that indicates the rank of the node and affects the DIO control message. It is a scalar representation of the location of the node in the DAG. Rank is used to avoid loops as well as to detect loops. The rank is not a path metric and it monotonically increases as the nodes go away from the root. As the node is away from the root its rank is increased.

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

The MinHopRankIncrease determines the maximum number of hops. The node checks the rank of the parents with the neighbouring node. Whichever has the lowest rank becomes the parent of that node. If both are equal then no change is made [08]. The rank is used to avoid loops and the routing metrics are used to find the shortest path between the nodes. When there are multiple roots, the node with the smallest rank is chosen as the preferred parent [05].

The Zero objective function uses the hop count as the routing metric to determine the rank of the nodes. Each node is assigned a rank based on the calculation made with the hop count.

$$R(N) = R(P) + Rank_{Increase} \quad (2)$$

where R(N) = Rank of the node and R(P) = Rank of the Parent Node.

$$Rank_{Increase} = (Rf * Sp + Sr) * MinHopRankIncrease \quad (3)$$

where Rf is the Rank Factor, Sp is the step of the rank and Sr is the stretch of the rank [10] [05].

In the minimum rank with hysteresis objective function (MRHOF) the expected transmission count (ETX) is used as the routing metric and the same is also used for the path metric calculation and determination of the rank. There is slightly a different approach from the OF0 is used here in the approximation of the rank for the nodes and the preferred parent. The node with the lower rank is not immediately chosen as the preferred parent, lest it creates a churn in the network. Whereas a threshold is set and if the rank is less than the set threshold then the switch over of the parent takes place. Otherwise the node continues to have its own parent in spite of the available parents with the lower rank [11].

B. Cooja Simulator

Cooja is a network simulator designed for simulating the sensor networks over the Contiki sensor Operating system. It is a Java based simulator but allows sensor nodes to be written in C [06]. Cooja is a flexible, cross-level simulator, which allows the nodes to be in different levels of not only software but also hardware. Cross-level simulation allows the simulation to take place at different levels of the system. Cooja combines both low-level simulation and high-level simulation. Cooja is not only flexible but also extensible to

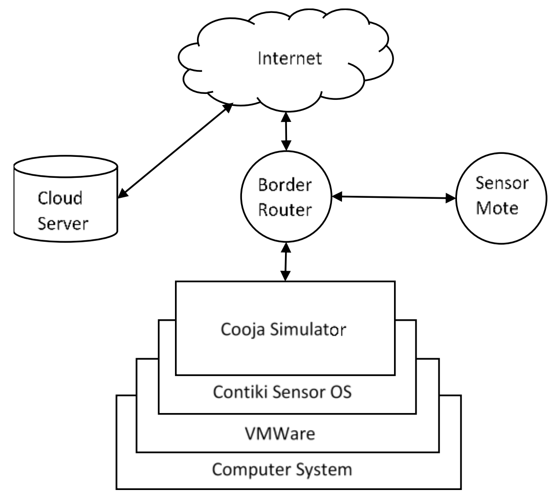

Figure 4. Cooja Simulator and Sensor Structure

different sensor node platform, operating system software, radio transceiver and radio transmission models. The other prevalent network simulators like NS2, TOSSIM, AVRORA and others are capable of running the simulation at only one level of the system. For example, NS2 can run only at the network level. The TOSSIM can run at operating system level and AVRORA at machine code instruction level, whereas Cooja can run at all these three levels at a time [12]. Fig. 4 explains the structure of Cooja simulation and its relationship with the sensor motes, internet and cloud server.

1) Working of ContikiRPL

The border router is initially set up by the user and it starts functioning as the root node. The root usually takes the ID number 1 and it sends out the DIO message to the neighbours, advertising its parameters. The rank of the border router will be the minimum, that is 1. The neighbours then calculate the rank for themselves and forward the messages. Any node with the lower rank is preferred as the parent by the subsequent nodes. The routing metric is used for the calculation of the rank, and it is determined by the objective function. This process ends once all the available nodes join the DAG. If the nodes do not receive the DIO within a specified time, then they stand sending the DIS message. The nodes that received DIS message would immediately transmit the DIO message.

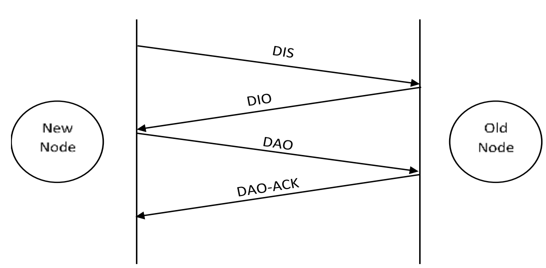

Figure 5. RPL Parent Selection

When all the nodes have joined the DAG, then they are ready for upward traffic and they send DAO to their parents. The parent node receives the DAO and updates its routing table and enables downward traffic. The parent selection process by the new nodes that are not part of the network also takes place in the same manner [06]. Fig. 5 illustrates the parent selection process for the new nodes in RPL [13].

2) Contiki Simulation Setup

The features and the advantages of Contiki Cooja Simulator over the other popular simulators had prompted us to choose it. We have performed the simulation with a single sink and a random network topology, in order to distribute the nodes of the network in the chosen area. We have selected a square area with 1000 square meters. The inbuilt and default OF0 objective function is chosen for the sample simulation. We have chosen 100 percentage for both the TX and RX success ratio. We are fixed to the default Imin values and the redundancy factor (k) value of the RPL. In order to capture the traffic, the radio messages are enabled and captured as pcap file. Later, the saved pcap file can be used to analyse the network using Wireshark network analyser.

TABLE I. CONTIKI COOJA SIMULATION ENVIRONMENT

Parameters	Value
Objective Function	OF0
Number of Motes	30
Topology	Random
TX Ratio	100%
RX Ratio	100%
TX Range	100m
Simulation Time	15 minutes
Squared Area	1000 meters
Wireless Channel	UDGM: Distance Loss

C. *Wireshark Network Analyser*

Wireshark is one of the most powerful and open source network analyser available. It is used to capture the network traffic and to inspect closely what happens in the network. It has a user-friendly and configurable GUI with many features. It can decode over 400 protocols and supports more than 750 protocols. New protocols are being added and it is actively being developed and maintained. It can run on more than 20 platforms like Unix, Windows and Mac OS. It works on promiscuous and non-promiscuous mode. It has the ability to both capture the network as well as to read the captured file. It has got rich filter display capacity. With all these features Wireshark stands tall among its counterparts like WinDump, EtherPeek, Tepdump, Snoop, Snort, Dsniff, Ettercap, Packetlyzer, MacSniffer and so on [14]. The pcap file contains the captured traffic details and the pcap file is captured by the Cooja simulator. Analysis of this pcap file provides a plethora of information about the network, link, node and the packets.

II. QOS MEASUREMENT PARAMETERS FOR RPL

The capability of a network to provide higher ranking performance and service is considered as quality of service (QoS). The performance of any network and the routing protocol can be measured through this QoS. In RPL, we take some of the outstanding parameters like ETX, latency, throughput, power consumption, convergence time and packet delivery ratio (PDR) as QoS measures. The main aim and objective of QoS is to give a guarantee that the network would provide the expected result. These parameters also give us the assurance that the network is reliable and would give the desired result [15].

A. *Expected Transmission Count (ETX)*

Expected Transmission Count is the number that specifies that number of transmissions of packets, a node expects from it to the destination successfully. The ETX is a discrete value computed from the following:

$$ETX = \frac{1}{Df * Dr} \qquad (4)$$

where, Df is the measure probability of a packet to be received by the neighbour and Dr is the measure probability that an acknowledgement packet is successfully received. ETX is one of the measures to determine the reliability of the link. The lower the ETX the better the reliability of the link and thus the quality of the link. ETX also must not exceed a specified limit [09].

B. *Latency*

Latency is a total delay of a packet starting from the moment of its release in the UDP layer up to its successful reception at the destination [16]. The latency can be calculated as a difference between the time when the packet was sent from the source and time when it was received at the destination. Thus, the total latency can be calculated for all the packets by summing separately the total received time and the total sent time and then finding the difference. From this we can derive the average packet delay or the average latency by dividing the total latency by the total number of packets received [17].

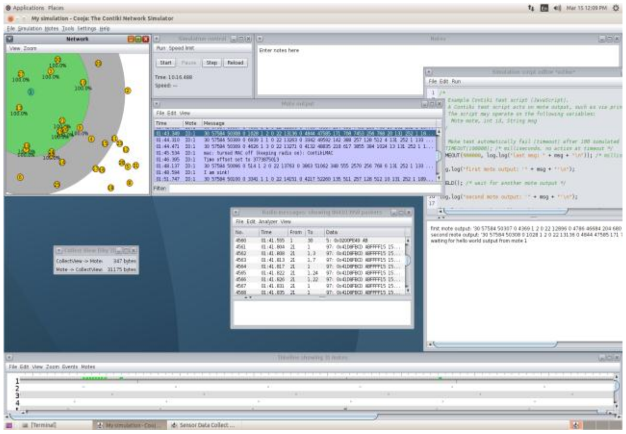

Figure 6. Cooja Simulator GUI

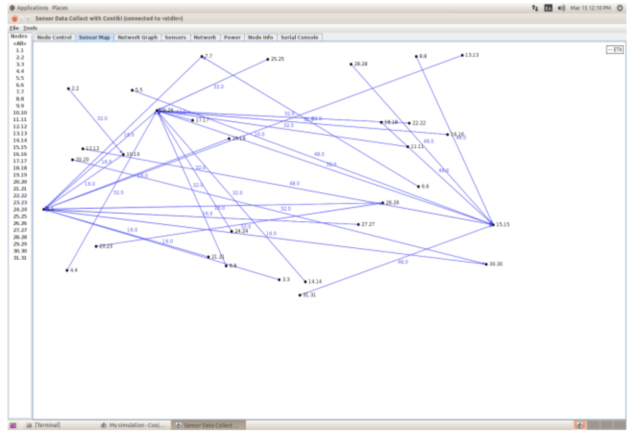

Figure 7. Sensor Map in Cooja Simulator

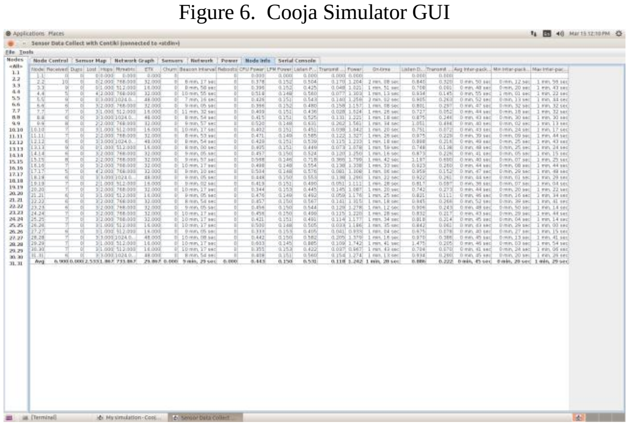

Figure 8. Collect View of Cooja

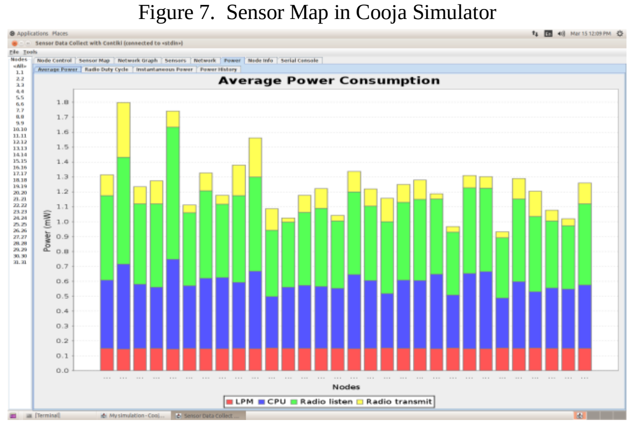

Figure 9. Power Consumption Histogram in Cooja

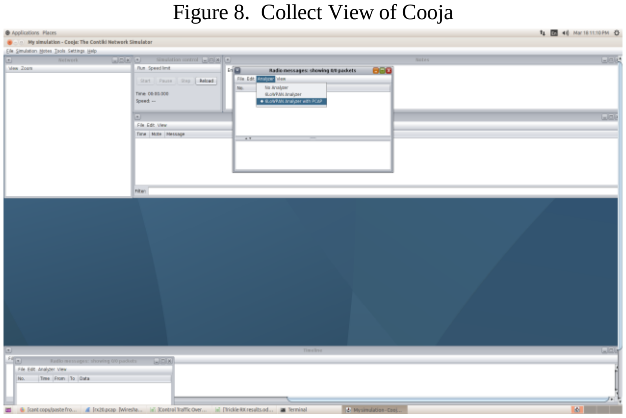

Figure 10. Enabling Capture in Cooja

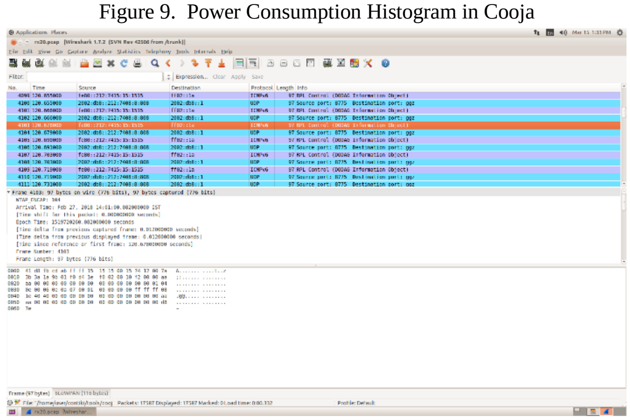

Figure 11. Viewing Captured File in Wireshark

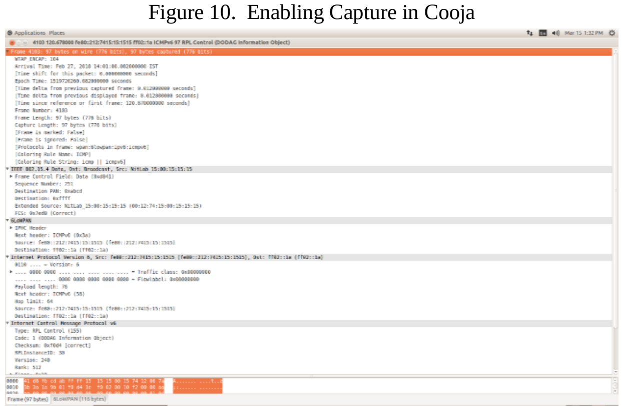

Figure 12. Packet Analysis View in Wireshark

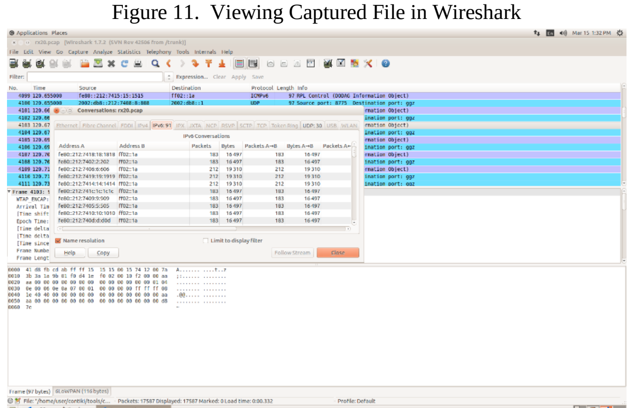

Figure 13. Filtering Conversation in Wireshark

$$Total\ Latency = \sum_{k=1}^{n} \left(Received\ Time\ (k) - Sent\ Time\ (k)\right) \quad (5)$$

$$Average\ Latency = \frac{Total\ Latency}{Total\ Packets\ Received} \quad (6)$$

C. Throughput

RPL considers throughput as one of the routing metrics. The throughput is also calculated like latency, from the start until end of the traversal of the packet through the link. Without entering into the complex details of the definition, we can abstractly define throughput as the product of the number of packets, the size of the packets and the integer 8 in order to convert the bytes into bits, divided by the total simulation time in seconds. This generalized formula is made due to the subtlety of throughput and the difficulty in measuring them. Throughput is dependent on the traffic workload of the network [18].

$$Throughput = \frac{(\,No. of\ delivered\ packets * Size * 8\,)}{Total\ duration\ of\ Simulation} \left(\frac{bits}{sec}\right) \quad (7)$$

D. Energy Consumption

The key critical issue in low power and lossy networks is energy consumption. They are low powered and energy conservation and efficient use of energy is vital to it. When we say energy not only the energy that is consumed is taken into account but also the residual energy that remains in the energy source. Mainly there are three sources of energy, namely, i) Mains-powered ii) Primary batteries iii) Energy scavenger. The energy scavengers may work well in LLNs, because there is a periodic stream of power input. The battery powered energy sources are less reliable comparing with the scavenger, because it is prone to be complete drain of energy. The power and energy values are very much dependent on the cost of sending and receiving the packets in the network. We use a single parameter in the Contiki Cooja to measure the energy consumption and the residual energy in battery powered energy sources. For the batteries we can calculate the average life time, based on its usage of energy consumption.

$$Ee = \frac{Eb}{E0^{\frac{(T-t)}{T}}} \quad (8)$$

where *Ee* is the Remaining Energy, *Eb* is the energy in battery, *E0* is the initial energy, T is the total life time and t is the total elapsed life time. The remaining energy is easy to be calculated that how the energy is being spent. But there may be a larger energy source that could store more energy than the one with tiny energy source. We cannot compare both, using the same scale of measurement. This method is useful only in the networks where similar type of energy storage is available [09].

1) Duty cycling

The LLNs have scarce energy resources and keeping the radio on all the time would drain the energy. Therefore, the radio is kept off as much as possible and turned on only when needed, in order to conserve energy. This method is called duty cycling. It can greatly reduce the energy consumption. There are different types of listening. One is idle listening, a method of listening to the idle channels long as it remains empty and until a packet it transmitted. It is an expensive duty cycling method. There are other techniques like sampled listening and scheduling. In sampled listening the channel is periodically checked for transmission, in scheduling only at a specified time [06].

Powertrace plugin is used in Contiki to trace the energy consumption. It measures the CPU energy, Low Power Mode (LPM) energy, Radio Transmit energy and Radio Listen energy. The CPU energy is the total energy used by the CPU and low power energy is the energy used by a node when it is in power saving mode. Both radio transmit and radio listen energies are the energies used by nodes to send and receive packets. The total energy used is the sum of all these four types. These parameters are internally measured by Contiki at run time and displayed concurrently [19].

E. Network Convergence Time

Topology formation is the important routing function of the RPL before it starts the transmission of data. The network needs to be set first before its starts transmitting data, therefore the network set up time is crucial in a network. DODAG structure is formed by the RPL, by sending control messages from the root. Until the end of topology formation or construction of DODAG a lot of control messages will be sent across the network. Convergence time is the total duration between the first control message and the last control message. Shorter convergence time renders more stability to the network [20].

$$Convergence\ Time = Last\ DIO\ joined\ DAG - First\ DIO\ sent \quad (9)$$

F. Packet delivery Ratio (PDR)

Packet delivery ratio of a network is the ratio between the total number of packets received by a node and the total number of packets sent to that node. The PDR value can be obtained by dividing the total number of packets received and the total number of packets sent. This value predicts the network reliability. The more the value of PDR, the higher the reliability of the network. At the same time another reliability metric ETX is inversely related to the PDR. If the value of the PDR is high then the value of ETX would automatically be very low [14]. We cannot have an absolute convergence time, if the nodes are mobile, but only an initial convergence time.

$$Packet\ Delivery\ Ratio = \left(\frac{Total\ Packets\ Received}{Total\ Packets\ Sent}\right) * 100 \quad (10)$$

G. Control Traffic Overhead

The control messages like DIO, DIS, DAO are being generated in RPL in order to setup the network and to maintain it. These control messages are absolutely necessary for the creation of DODAG. Control traffic overhead is the total sum of all types of control messages in the network. The efficiency of the routing protocol depends on controlling the number of these messages keeping in mind the scare energy resources in IoT. At the same time the reduction of the control messages is challenging if the network is on constant flux. RPL makes use of the trickle algorithm to reduce the control traffic overhead [14].

$$Control\,Traffic\,Overhead = \sum_{k=1}^{m} DIO\,(k) + \sum_{k=1}^{n} DIS\,(k) + \sum_{k=1}^{o} DAO\,(k) \quad (11)$$

III. METHODS OF MEASURING THE QOS PARAMETERS

Cooja simulator has a user-friendly and easy to configure GUI as shown in Fig. 6. There is a control panel to control the traffic. The environment can be set and the motes, types of motes, the radio environment, the transmission range, mote ID and many other details can be viewed online in the simulator. The sensor network will appear as seen in Fig. 7. The collect view is shown in Fig. 8, which gives around 21 node metrics quantities. The four types of node energy consumption are graphically displayed in Fig. 9. The pcap file could be enabled as we see in Fig. 10. The Wireshark network analyser is used to read the captured pcap files. Fig. 11 shows the view of pcap file opened in Wireshark. The analysis of the packet is shown in Fig. 12 and Fig. 13 shows the conversation filtering in Wireshark.

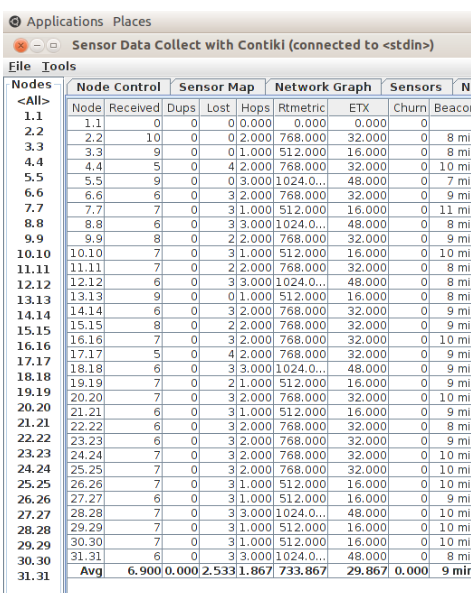

Figure 14. ETX Value Displayed in Collect View

A. Measurement of ETX

The collect view of the Cooja simulation provides the online and updated data of the ETX values. After the start of the simulation, the collect view can be opened. The node information of the collect view would provide the online ETX value. After the simulation at the end of the stipulated time, the ETX value should be collected. We can take the values, individually for each node or take the average of ETX value of the whole network. Fig. 14 shows the collect view output of the ETX in Cooja simulator. The ETX value will vary according to the duration of RPL simulation. The lower the value of ETX the better the reliability of the protocol.

B. Measurement of Latency

The time delay of the packet transmissions can be calculated from the pcap file analysed in Wireshark. Each transmission has the time as well as the mote ID and the difference between the sending time at any node and the time of reception of the packet at the sink can be calculated for each node. The sum of the total would give the total latency and the average of the time will give average latency. Like ETX, the end to end delay has to be minimum for any network to give a better performance.

C. Measurement of Throughput

The throughput can be calculated by analysing pcap file. The total number of packets transmitted and the total time of simulation. Otherwise it is also easier to find out from the summary statistics in the Wireshark application, which is given in Fig. 15. The throughput value should be more for a network, in order to be considered better.

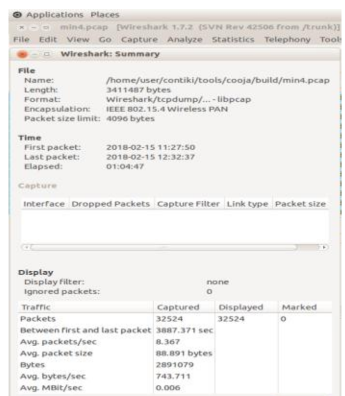

Figure 15. Summary Statistics in Wireshark

D. Measurement of Energy Consumption

The collect view of Cooja simulator provides the tabulated details of the energy consumption of each individual node in all the four levels, namely CPU energy, LPM energy, Radio on time Energy, Listen time energy. The figure. X shows the display of the tabulated energy consumption. It also possible to find the four levels of energy consumption individually and collectively using this GUI display. The energy consumption output is shown in Fig. 9. The lower the energy consumption of a node and the network, the better it suits for the LLNs.

E. Measurement of Convergence Time

The pcap file captured in Cooja is evaluated by Wireshark network analyser. We use the filter to select only the DIO messages and find the first and last DIO control message. We can also get the convergence time by analysing the mote output of Cooja simulator as shown in Fig. 16. The mote output provides details of the time at which the message is sent, the mote ID number and the details of the transmission. It is very easy to locate the last DIO that joined the DAG in that mote output file. The convergence time needs to be the minimum for a network to provide better stability. Usually it takes from 5 seconds to 15 seconds for a network to get converged. Depending on the mobility of the nodes it may vary.

F. Measurement of PDR

The packet delivery ratio can be calculated using the pcap

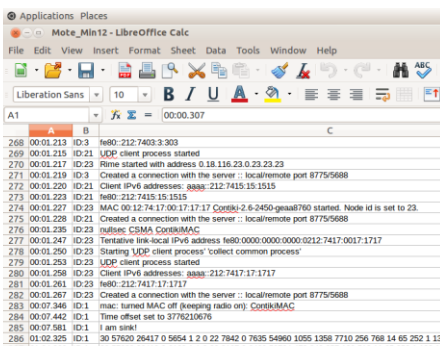

Figure 16. Transmission Time in Mote Output

file in the Wireshark network analyser. The total sent packets and the received packets can be filtered using the filtering mechanism in the Wireshark. Packet delivery ratio is one of the main factors in measuring the reliability of a network. A network with a good transmission range will provide more than 90 % of PDR. The network size affects the PDR value.

G. Measurement of Control Traffic Overhead

We take into consideration once again the pcap file for the analysis of the control traffic overhead. We filter out only the DIO, DIS and DAO control messages. The sum of all these provides the total control traffic overhead of the network. The control overhead should be reduced as much as possible, because more traffic would drain the batteries and affect greatly the low powered devices in LLN.

IV. CONCLUSION

The quality of any network consists in the way it delivers the expected result. The quality of services guarantees the network quality and performance. We have taken into consideration some quality measures to ensure the quality of RPL. The results derived from using these measurements can be of great help to predict the robustness, reliability, stability, resilience and other vital qualities of the network.

REFERENCES

[1] C. Thomson, I. Wadhaj, I. Romdhani and A. Al-Dubai, "*Performance evaluation of RPL metrics in environments with strained transmission ranges*," 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), **Agadir**, pp. **1-8**, **2016**.

[2] Yaqoob et al., "*Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges*," in IEEE Wireless Communications, Vol. **24**, No. **3**, pp. **10-16**, June **2017**.

[3] A.W. Atalay, "*Design and Analysis of Routing Protocol for IPv6 Wireless Sensor Networks*", Diss. University of Pisa, **Italy**, pp. **5-30**, **2015**

[4] B. Daniel., "*A Performance Evaluation of RPL with Variations of the Trickle Algorithm*", Diss. Worcester Polytechnic Institute, pp. **6-20**, **2016**.

[5] W. Mardini, M.Ebrahim, M.Al-Rudaini, "*Comprehensive Performance Analysis of RPL Objective Functions in IoT Networks*", International Journal of Communication Networks and Information Security (IJCNIS), Vol. **9**, No. **3**, pp. **323-332**, Dec **2017**.

[6] H. Ali, "*A Performance Evaluation of RPL in Contiki*". Master's Thesis, Blekinge Institute of Technology, **Sweden**, (ELnaz), pp. **5-85**, **2012**.

[7] "*Critical Evaluation of RPL Routing Protocol for WBAN*", Chapter 4, **India**, pp. **79-100**

[8] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struick, J. Vasseur, R. Alexander, "*RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*", IETF RFC 6550, **2012**

[9] J. P. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthe, "*Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks*", RFC 6551, Internet Engineering Task Force RFC 6551, March **2012**.

[10] P. Thubert (Ed), "*Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)*", RFC 6552, IETF, **2012**.

[11] O. Gnawali, P. Levis, "*The Minimum Rank with Hysteresis Objective Function*", IETF RFC 6719, **2012**

[12] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne and T. Voigt, "*Cross-Level Sensor Network Simulation with COOJA*," Proceedings. 2006 31st IEEE Conference on Local Computer Networks, **Tampa**, FL, pp. **641-648**, **2006**.

[13] T. Mehmood, "*COOJA Network Simulator: Exploring the Infinite Possible Ways to Compute the Performance Metrics of IOT Based Smart Devices to Understand the Working of IOT Based Compression & Routing Protocols*", Cornell University, Islamabad, pp. **1-7**, **2007**.

[14] A. Orebaugh, G.Ramirez, J. Burke, L.Pesce, J. Wright, G.Morris, "*Wireshark & Ethereal Network Protocol Analyser Toolkit"*, Syngress Publishing, Inc., Rockland, pp. **13-53**, **2007**.

[15] M. Asif, S. Khan, R. Ahmad, M. Sohail and D. Singh, "*Quality of Service of Routing Protocols in Wireless Sensor Networks: A Review,*" in IEEE Access, vol. **5**, pp. **1846-1871**, **2017**.

[16] M. Vučinić, B. Tourancheau and A. Duda, "*Performance comparison of the RPL and LOADng routing protocols in a Home Automation scenario*," 2013 IEEE Wireless Communications and Networking Conference (WCNC), Shanghai, pp. **1974-1979**, **2013**.

[17] N. Pradeska, Widyawan, W. Najib and S. S. Kusumawardani, "*Performance analysis of objective function MRHOF and OF0 in routing protocol RPL IPV6 over low power wireless personal area networks (6LoWPAN)*," 2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE), Yogyakarta, pp. **1-6**, **2016**.

[18] S. A. Jyothi, A. Singla, P. B. Godfrey and A. Kolla, "*Measuring and Understanding Throughput of Network Topologies*," SC16: International Conference for High Performance Computing, Networking, Storage and Analysis, Salt Lake City, UT, pp. **761-772**, **2016**.

[19] I. N. R. Hendrawan and I. G. N. W. Arsa, "*Zolertia Z1 energy usage simulation with Cooja simulator*," 2017 1st International Conference on Informatics and Computational Sciences (ICICoS), Semarang, pp. **147-152**, **2017**.

[20] Nguyen Thanh Long, N. De Caro, W. Colitti, A. Touhafi and K. Steenhaut, "*Comparative performance study of RPL in Wireless Sensor Networks*," 2012 19th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT), Eindhoven, pp. **1-6**, **2012**.

Authors Profile

A.S. Joseph Charles is a Full-time Research Scholar in the Department of Computer Science and Applications, The Gandhigram Rural Institute (Deemed to be University). He received his Bachelor degree in Computer Science from Madras University, Master degree in Computer Applications from The Gandhigram Rural Institute (Deemed to be University). He has 8 years of teaching experience in Community College and Arts and Science College. He has also received his Bachelor degree in Philosophy from Salesian Pontifical University, Rome and Bachelor degree in Catholic Theology from University of Rome. The author is interested in the areas of Internet of Things and Android applications Development.

P. Kalavathi, Associate Professor and Head of the Department of Computer Science and Applications, The Gandhigram Rural Institute (Deemed to be University), obtained her BCA degree from Mother Teresa Women's University, Kodaikanal, Post-Graduate Degree in Computer Applications (MCA) at GRI-DU, M. Phil in Computer Science from Bharathidasan University, Trichirappalli and Doctoral Degree in Computer Science and Applications from GRI, Gandhigram. She was qualified in UGC-NET for Lectureship in 2000. She has 17 years of Teaching and 10 years of Research Experience. The Author has more than 50 research articles in various journals and edited volumes. Her research area focuses on Internet of Things, Digital Image Processing, Medical image Segmentation & Analysis and Medical Image Compression. She is also serving as a reviewer of many international Conferences and various journals in IEEE, Springer, Elsevier etc. The author is a Life Time Member of Indian Society for Technical Education (ISTE), New Delhi.

