RESEARCH ARTICLE OPEN ACCESS

Methods for Assessing the Power Consumption of IOT Routing Protocols

Janappa R. Hugar
Selection Grade Lecturer
Electronics and Communication Engg Department
Government Polytechnic, Bagalkote. Email: jrhugar73@gmail.com

ABSTRACT
The Internet of Things (IoT) is a game-changing innovation that will affect every facet of our daily lives. The interest in IoT technologies is astronomical, as seen by the rapid expansion of unique, low-cost sensor and actuation devices in recent years. Device limitations, such as those affecting power, storage, bandwidth, and computing capability, create several difficulties for IoT devices. If you want your battery powered IoT gadget to last as long as possible, you need to make sure it has a low power consumption design. The sending and receiving of data packets consumes most of the power in IoT networks. Maintaining reliable transmissions while reducing power consumption is difficult. To get these packets from their origin to their destination, routing protocols are crucial. Using routing protocols that are gentle on power usage is one way to lessen the burden IoT devices have on the grid. Various routing methods have been created to facilitate data transmission via networks. To choose a protocol that works well in low-power Internet of Things settings, it is necessary to compare available options. Lightweight On-demand Ad hoc Distance-vector Routing Protocol-Next Generation (LOADng) and RPL (Routing Protocol for Low Power and Lossy networks) are compared in terms of their energy consumption in this research. The energy used by IoT devices in a network during data transmission is the primary subject of this research. The Cooja Network Simulator is used to test the Routing protocols once they have been implemented in the Contiki OS. The findings show that the RPL routing protocol has a low energy footprint.
Keywords: IoT, LOADng; RPL, energy-efficiency; Cooja

I. INTRODUCTION

Optimal network performance on the Internet of Things relies on energy efficiency. Scalability, information and knowledge management, ubiquitous data exchange, optimized energy solution, self-organization, and localization track capabilities are just some of the hallmarks of IoT technology, which communicates with a wide range of physical devices (device heterogeneity). Due to the node's restricted communication range, the collected data is sent to the intermediate node. High energy consumption for nodes is caused by the intermediate node's packet forwarding of the source node, which in turn promotes network partitioning [1, 4]. Numerous routing protocols, including the non-standard extension of RPL called Cognitive and opportunistic Routing Protocol for Low power and Lossy Network (CORPL)[6] that is tailored for cognitive networks and makes use of opportunistic forwarding to forward packets at each hop, the Collection Tree Protocol (CTP), RPL, and LOADng, have been introduced on the Internet of Things (IoT) setting.

The nodes can communicate with one another thanks to the control messages used in these protocols. The LOADng routing protocol is a reactive routing system for LLNS that is based on Ad Hoc OnDemand Distance Vector (AODV). When two nodes desire to exchange data messages with one another, a route is created in LOADng, just as it is in AODV. LOADng is another resource-intensive routing mechanism. The development of routes is carried out by Control messages, which make use of a collection of data held by nodes throughout protocol execution. Using the current goal function (hop-count and ETX), the IETF (Internet Engineering Task Force) group has standardized the Routing Protocol for Low power and lossy networks (RPL) [2]. Since there are so many wireless devices constantly exchanging data with one another, routing is one of the most pressing concerns on the Internet of Things. Without intelligent forwarding and routing performed by routing protocols, significant packet loss and retransmission would occur, resulting in higher energy, memory bandwidth, and processing capacity consumption [8]. For IoT networks to last as long as possible, the right routing algorithm that cuts down on power consumption is essential.

As a result, the energy efficiency of RPL and the LOADng protocol are compared in this investigation. Data transmission in a dispersed network is the primary focus of this investigation on node power usage. The remaining parts of the paper are structured as follows. The associated work is briefly described in Section II. Explain the routing protocols in Section III. In Section IV, we discuss how we set up our simulation. Evaluation of the experimental results is discussed in detail in Section V. Section VI concludes with some last thoughts and a look toward the future.

II. RELATED WORK

Establishing routing protocols is one method of reducing the energy consumption of IoT devices. The authors of [3] gave an analysis of the usefulness and performance of the routing protocols RPL and LOADng in AMI networks and LLNs. Protocols' restrictions on AMI network traffic patterns: Low-delay RPL designed specifically for sensor-to-root communications. Unlike LOADng, which is compatible with a wide variety of traffic patterns, RPL has significant drawbacks and is not appropriate for networks with unidirectional connections. The goal of the researchers that introduced the energy-efficient routing protocol for IoT applications in [5] was to create a system for IoT applications that uses less energy since the nodes require less electricity.

According to the results, the node determines the most efficient level of transmission power at which to send packets. Due to the test-bed scenario's need for homogeneous nodes, they were only able to simulate two nodes, but they recommended that the system could function with a bigger number of nodes. For effective data acquisition in LLNs, the authors of [6] offered an assessment of the LOADng-CTP and RPL routing protocols, as described by an IETF draft enhanced with a collection tree. This research compared the performance of MP2P and P2MP routing protocols by measuring their control overhead, End-to-End Delay, and Packet Delivery Ratio.

Contiki OS's built-in protocols. According to the findings, LOADng-CTP offers superior performance to LOADng in terms of packet delivery ratio, delay, and overhead, and is backwards-compatible with RPL in terms of these same metrics for bidirectional data traffic, making LOADng-CTP the superior solution for LLN networks. For dynamic power profiling, the authors of [7] describe a component-based design that makes use of a power estimating module.

The plan was put into action and tested in a WSN environment using TelosB nodes as the smart objects and the Contiki operating system. The power is determined by calculating the total of the power used by the node's transmitters, receivers, central processing units, and local power management modules. To determine the performance assessment routing metrics used in LOADng, considering a variety of traffic patterns and network sizes, the authors of [8] give an evaluation of these measures. Data Analysis

Statistical measures of packet transmission and reception, including delay, energy consumption, and network topology. Ninety percent of a sent packet is shown in the findings for lesser networks. As a result of using the most dependable route, the measurements guarantee a high packet delivery ratio. Metrics demonstrate that the LOADng protocol's performance degrades with the number of hops used to find a trustworthy route increases. As energy usage rises, network efficiency drops. Increasing the network's longevity via energy efficiency is a primary focus. The purpose of this research is to identify the most power-efficient routing protocol for LLNs by comparing the power requirements of existing routing protocols. Both the LOADng and RPL protocols will be tested in large and small networks.

III. ROUTING PROTOCOL

The Routing protocol is crucial to the smooth operation of any network. The job of routing is carried out by routing protocols in IoT dispersed networks.

A) Protocol for routing in networks with limited power and high loss.

RPL was developed at IETF as a distance-vector protocol for use in IoT routing. The RPL's primary function is to stand in for a tailored solution in low-power, lossy networks; it was designed to be very flexible in response to changes in the underlying network and to provide backup paths if the primary ones become inoperable [4].

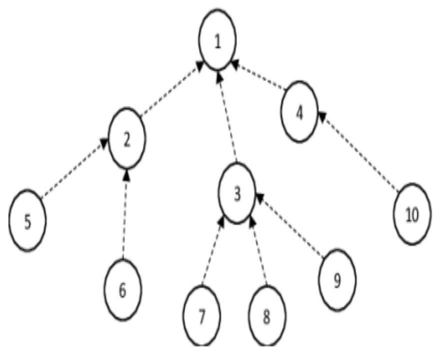


Fig No.1 – RPL Tree Topology

The control messages in RPL are what are utilized to establish the infrastructure of a network. The protocol generates DODAGs, which are directed acyclic graphs with just one path from each leaf node to the root, the destination of all communication leaving that leaf. As can be seen in picture 1, RPL generates a topology that resembles a tree, complete with a central root and radiating branches and leaves. The protocol's control messages are used to transmit data packets, receive data packet acknowledgements, and transmit data packets.

Each node in the network broadcasts a DODAG information object (DIO) claiming to be the tree's central hub. As DIO is shared throughout the network, the whole DODAG is constructed. A node's communication begins with the transmission of a destination advertisement object (DAO) to its parents, which is then sent to the root, which ultimately selects which path to take [3]. Sending a DODAG information solicitation (DIS) is the first step for new nodes entering the network and receiving a DAO acknowledgement (DAO-ACK) from the root is confirmation that they have been accepted. Nodes in an RPL network may be either stateless (the default) or stateful. A stateless node can remember the DODAG of its parent node. Therefore, the root is the hub of all communication. Since a stateful node remembers its offspring and ancestors, it may avoid going via the DODAG's root node when interacting with other nodes in the same subtree [11].

For effective network interaction, the protocol makes use of objective functions, routing metrics, and routing restrictions. Path optimization is achieved via the use of the objective function Minimum Rank with Hysteresis Objective Function (MRHOF), which allows nodes to discover the least expensive pathways while limiting network churn and keeping power consumption constant [5]. The goal function keeps the lights on by determining the least-direct route, which, if shorter than the present one, is taken instead. Control messages in the protocol are routed using ETX (Expected transmission count) and Hop count metrics. Hops builds a route with a minimal number of hops regardless of the ETX values of the connections, while ETX builds a path with a minimum expected number of transmissions regardless of the number of hops necessary to reach the destination. A tree is constructed by broadcasting RPL, DAG, and DIO messages from one node to another until the message reaches its destination. Based on the node's rating, it selects the best possible linkages and hops. When determining the optimum route, RPL prioritizes low latency and low power consumption.

B) LLN Next-Generation On-Demand Ad hoc Distance-vector Routing Protocol.

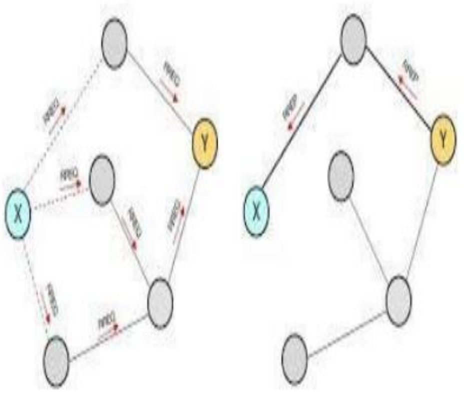


Fig No. 2 : LOADng operation RREQ and RREP

Lossy, low-power networks Minimalist On-Demand In order to transmit, acknowledge, and receive packets, the Ad hoc Distance-vector Routing Protocol relies on control messages. Intermediate nodes simultaneously learn a forward route towards the target and add this knowledge to the routing tables, while other nodes relay Route Reply (RREP) messages via the already established reverse route. When RREP is deployed to intermediate nodes along the request's bidirectional journey, it eventually reaches the node from whence the request was initiated. The detecting node may attempt to repair the connection by broadcasting an RREQ on behalf of the data source node; if route repair fails, the node unicasts a route error (RRER) message procedure and it updates its routing databases. The acknowledgement of a Route (RREPACK) Produced by a LOADng router after receiving an RREP to let the sending neighbor know that the RREP was received. Because LOADng takes a reactive approach, intermediate nodes are not permitted to send an RREP message even if they know the path to the destination. Each LOADng node needs its own Information Base to keep track of routing data and other network node details [9]. When determining how to go from one node to another, LOADng prioritizes the measure of hop count. The optimal route is one that minimizes the total number of hops while maintaining high reliability of connections and low power consumption, and this is represented by the hop count.

C) CoRP, CARP, and COAP

In [12], the authors propose a cognitive machine-to-machine (M2M) routing protocol (CoRP) for the Internet of Things that combines the features of a centralized cognitive medium access control (MAC) protocol with those of a distributive cognitive M2M protocol, with a focus on M2M routing. This protocol offers faster throughput and reduced latency, and it has been shown to function explicitly in cognitive radio enabled applications. In [13], researchers suggested a multipath routing protocol called Congestion Avoidance Multipath Routing Protocol (CA-RPL) to improve upon the standard routing protocol for low-power, lossy networks (RPL). The protocol is to blame for the overloaded networks. Congestion in the network is due to low-quality connections, heavy data flows, and packet drops. Directed acyclic graph (DAG) building times may be shortened thanks to CA-RPL's newly acquired knowledge of routing costs.

According to their findings, CA-RPL significantly improves upon the state of the art in RPL with regards to packet loss ratio and time delay. However, CA-RPL was not successful in achieving energy conservation at mobile or multisink nodes. The Constrained Application Protocol (CoAP) is introduced by the authors of [14]. The protocol was created with capabilities like asynchronous message exchange, etc., to ease communication between Machine-to-Machine (M2M) applications. It aims to design a web protocol on top of User Datagram Protocol (UDP) for unique environments with limited nodes and a subset of HTTP features that have been adapted to work with the limited resources of the embedded device. The protocol also allows for low-overhead multicast transmissions.

IV. SIMULATION SETUP

To implement, we employed the Cooja simulation [18]. The simulation was chosen because it represents a significant step forward for software used in wireless sensor networks. It's a simulator written in Java that can run code written in C. For this analysis, a simulated network was constructed with a single sink node, and nodes were distributed over a squared network architecture according to the Random topology. Both the RPL and LOADng networks (5, 10, 15, 20, 25, 30, and 35 nodes) were used in the trials, however their densities varied.

The time for simulation was specified in the Simulation script editor before the simulation was run. We have a 100% TX range, a 100% TX ratio, and an 80% RX ratio. The RX power was set to 80% so that the node may make use of its neighbors' listening capabilities and prevent packet loss and duplication. We tried increasing the TX ratio to 100%, but the nodes were not reacting quickly enough to the huge volume of packets being forwarded at the same time since they were listening to far-off nodes instead of their neighbors.

To gather data from the nodes, the Sensor data collector was activated. Each node's power consumption during packet transmission and reception, duty cycle, and hop count are all provided by the sensor data collector. Energy consumption during packet transmission and reception is being measured across a variety of routing protocols in this project, with an eye toward understanding how different protocols affect a node's CPU and LPM use. The default values for the simulation variables are shown in Table I. The motes' physical configuration may be seen in the network pane, which is seen in figure 3 below. The motes' physical location might be altered to create a topology.

Each node in the network window is colored differently according on its role; for example, a sink mote will be green, whereas a transmitter mote would be yellow. The network windows also displayed mote properties, a radio environment of each mote, mote kind, and radio communication between the motes. The simulation speed, as well as pausing, restarting, and reloading the currently running simulation, may all be adjusted using the simulation control window. You may record your thoughts on the simulation's theory and its most salient findings in a notes window.

Parameter	*Value*
Operating System	Contiki 2.7
Routing protocols	RPL, LOADng
Mote type	Tmote Sky
Number of motes	5-35
Topology	random
Tx Ratio	100%
Rx Ratio	80%
Int range	100 m
Tx Range	50 m
Simulation Time (minutes)	10
Wireless channel	UDGM Distance loss

Table No. 1 : Simulation Reading

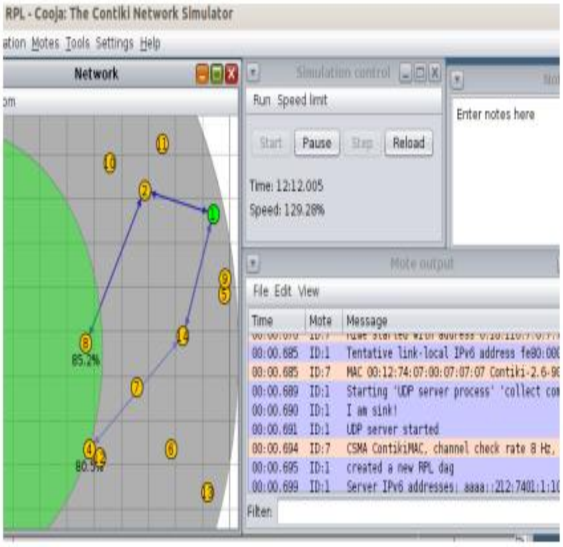


Fig No.3 - Simulation environment

The original DODAG tree used in the RPL simulation consisted of a single sink node at its root and four source nodes. The program udp-sink. c is running on the sink node, while udp-server. c is running on the client nodes. This simulation process sink node has been chosen to demonstrate its capabilities. Options for the mote's tools, transmission ranges, TX/RX, etc. may be seen and adjusted. Client nodes initiate the network and transmit packets to the sink node, which then collects the data. Each simulation ran for 10 minutes, and the results were recorded. The nodes' sensor data collector was activated, and it has begun collecting data. Duty cycle, network graph of each protocol, sensor map, etc. are all seen by a sensor data collector. this includes the power used by each node's central processing unit, local processing module, listen to, and transmit. Ten iterations of the simulation technique were performed on RPL for 5 nodes. Each network size between 5 and 35 nodes was subjected to the same method. Cooja, the same simulator used to simulate the RPL protocol, may be used to model the LOADng protocol network. Nodes were discovered using the route-discovery.c program included in the rime directory. The same procedure is used for both RPL and simulation. Each simulation's average power was determined by importing the sensor data collector's output into an Excel spreadsheet.

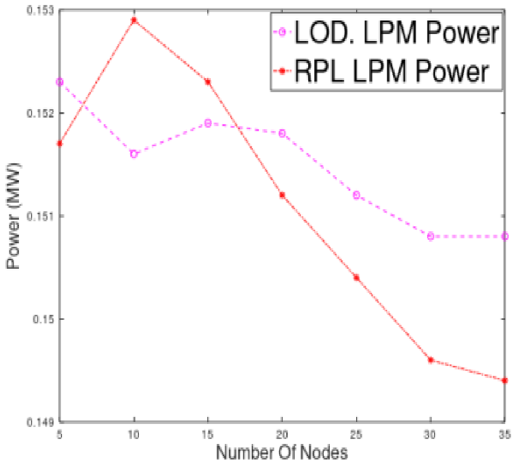
V. EVALUATION RESULTS

Here, we show how LOADng and RPL fared in an examination of their energy efficiency. Data from the sensors in the Cooja simulator's data collector was used to calculate the power usage. Statistical analysis of the energy efficacy of routing methods has been performed. The outcome of a simulation's evaluation of the node's power, determined by averaging the power of the node's constituent parts. There is processing power, memory, data transfer, and auditory reception [13]. To determine how much energy each protocol utilized on average to send and receive data packets, we used Microsoft Excel. The power data were easily interpreted graphically with the help of Octave 4.4.1, which also allowed for comparison of the routing protocols, energy efficiency, and suggestions.

Protocol	*No. of Nodes*	*CPU power*	*LPM power*	*Listen power*	*Trans power*	*TOTAL power*
RPL	5	0.3152	0.1517	0.3982	0.0961	0.9612
	10	0.3465	0.1529	0.4094	0.0756	0.9844
	15	0.3710	0.1523	0.4735	0.0915	1.0523
	20	0.4060	0.1512	0.4664	0.1025	1.1261
	25	0.4301	0.1504	0.4722	0.0864	1.1391
	30	0.4554	0.1496	0.49	0.0933	1.1883
	35	0.4702	0.1494	0.5131	0.1051	1.2378
LOAD ng	5	0.3668	0.1523	0.4045	0.0992	1.0228
	10	0.393	0.1516	0.504	0.1821	1.2307
	15	0.3872	0.1519	0.4887	0.17635	1.20415
	20	0.393	0.1518	0.5474	0.1536	1.2458
	25	0.4002	0.1512	0.6438	0.18	1.3752
	30	0.4164	0.1508	0.7418	0.2358	1.5448
	35	0.4171	0.1508	0.8173	0.2325	1.6177

All provided graphs reflect the mean performance in a set of 10 independent experiments. In order to assess and compare the power consumption among the routing protocols, the data acquired from the simulations is shown in graphical form in Table II above. Repeating the experiment ensures a more accurate average reaction time by canceling out the impacts of random fluctuations, such as a sudden increase in memory consumption from other running programs, which might slow down our investigation.There were two dimensions used in the analysis of power. The power of the nodes' LPMs, CPUs, transmitters, receivers, and other subsystems was first measured. CPU sleep mode power, or LPM, keeps the node using less energy when sending packets. TX is the transmitting power of the node, while RX is the receiving power.

To determine the overall energy required by the protocol, it is necessary to add up the individual power requirements of its many parts. Total power consumption equals the sum of each component power requirements. 1 Where P_CPU represents the active-mode power consumption of the node. In low power mode (abbreviated as P_LPM), the CPU consumes very little power. The energy expended by the nodes during packet transmission is denoted by the notation P_tx, or transmission power. P_rx, listening power, is the amount of energy used by a node when listening for incoming packets or processing replies to a request. The following graphs depict the outcomes of the tests that were conducted.The CPU use of the LOADng and RPL routing protocols in IoT-wide ad hoc networks is shown in Figure 4. As the number of nodes expands, so does the computing power available to it. The rise in CPU power indicates that the nodes' idle states were associated with higher power consumption. While RPL's CPU power steadily rises, LOADng's CPU power is erratic, fluctuating between decrease and increase. As CPU power rises, the ability of individual nodes to send data declines. As a result, RPL has more processing power than LOADng.



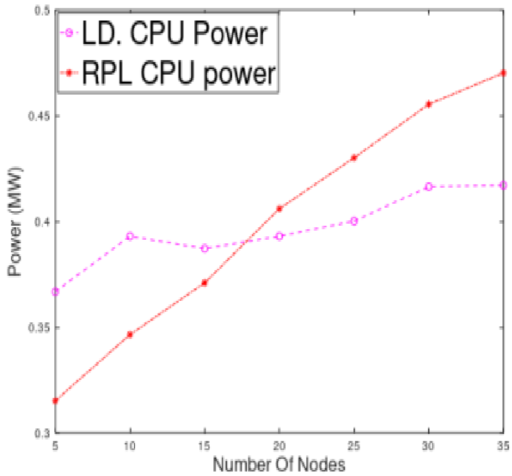


Fig. 4 LOADng and RPL CPU power

Increases in RPL LPM power are seen in Figure 5 above at smaller network sizes, whereas declines are shown at greater network sizes. The LPM power is the sleeping CPU's power, which influences transmission power and reception power. In contrast to RPL LPM power, LOADng LPM power rises as the number of nodes in a network grows, resulting in more energy expenditure while transmitting and receiving data. Power use in LOADng and RPL protocols during transmission (Tx) mode is shown in Figure 6. When nodes exchange data with one another, they use up energy. When a node creates a message, it consumes some amount of power. Sender nodes broadcast the packets as a request or convey a given message to the sink or receiver node. When transmitting packets from a source to a destination node, LOADng nodes use more energy than RPL nodes. More energy is used by the LOADng protocol as it searches for available transmission channels before actual packet transmission begins. RPL leverages strong networks and short hops to reduce transmission power consumption compared to the LOADng protocol.

VI. CONCLUSION

The goal of this research was to compare the Energy Efficiency of several Internet of Things routing protocols (such LOADng and RPL). Cooja simulator was used for the assessment, and the results are shown in figure 3. The simulation's nodes were randomly placed throughout the network. Hop count and ETX were used in the study of energy efficiency in routing. The hop count reveals the total number of relays required by a packet to reach its destination. According to the results of the trials, a larger number of hops indicates that the message traveled a longer distance to reach its destination, using a greater amount of energy. When comparing RPL with LOADng, RPL prioritizes using the shortest pathways possible. Transmission, reception, and node lifespan are all conserved by the Routing Protocol for Low power and lossy networks (RPL), but more energy is used by the LOADng protocol. The LOADng protocol's route-discovery mechanism causes it to have a high Duty cycle ratio, meaning that the protocol is always on and using power.

For maximum efficiency, it is preferable to create a route only when a node is ready to transfer data packets, as is the case with the RPL routing approach. To reduce energy consumption, nodes must be able to choose a trustworthy route to the destination before sending any packets. Power assessment in IoT routing protocols is a future area of research. Since LOADng uses more energy than RPL, it will not be included in the study's evaluation of network layer protocols. Packet analysis, energy use, and distributed network security will all factor into the research.

REFERENCE

[1]. Alvi, S. A, F. ul Hassan, and Mian, A. N, 2014 On the Energy Efficiency and Stability of RPL Routing Protocol, Wireless. Communication Mobile. Computing,2015 (pp.19271932).
[2]. A. Tiruchirappalli, A. Tiruchirappalli, and T. Nadu, 2015. NETWORKS, (vol. 7, no. 4, pp. 93100).
[3]. J., Yi, T., Clausen, and Y., Igarashi, 2013. Evaluation of routing protocol for low power and Lossy Networks: LOADng and RPL. 2013 IEEE Conference. Wireless. Sensor (ICWISE 2013, no.Lix, pp. 19 – 24,)IEEE.K. Elissa, Title of paper if known," unpublished.
[4]. A., Parasuraman, D., Culler and R., Katz, 2015 An Analysis of the RPL Routing Standard for Low Power and Lossy Networks. Technology, (Rep. No. UCB/EECS-2014-106, p. 98.).
[5]. M., Richardson, and I., Robles, 1994, RPL-Routing over Low Power and Lossy Networks. Presentation, 1994.
[6]. B., Mohammed, and F., Mohamed, 2014. Performance comparative analysis of LOADng-CTP and RPL routing protocols for LLNs, 2014 (vol.12, pp 12-16)
[7]. N.S., Kumar, S., Koshy and G.S, Kumar, 2015. Dynamic power profile estimation of interconnected smart objects in Intell. Computing System, 2015 IEEE. (RAICS 2015, no. December, pp. 394399, 2015).
[8]. J.V.V, Sobral, J.J.P.C, Rodrigues, N., Kumar, C, Zhu, and R.W, Ahmad. Performance evaluation of routing metrics in the LOADng routing protocol. Communication Software Systems., vol. 13, no. 2, pp. 8795, 2015.
[9]. I.D, Chakeres, and E.M, Belding-Royer, AODV Implementation Design and Performance Evaluation, International Journal Wireless Mobile Computing. IJWMC, vol. 3, no. 2, pp. 110, 2005.
[10]. A Latib, A.N, Jamil, A.M, Alduais, J, Abdullah, L.H.M, Audah, and R, Alias, 2015. Strategies for a better performance of RPL under mobility in wireless sensor networks, AIP Conference., 2015(vol. 1883).
[11]. A, Dunkels, 2011. The ContikiMAC Radio Duty Cycling Protocol, SICS Tech. Rep. T201113, ISSN 1100-3154, pp. 1 11
[12]. A, Dunkels, F, Osterlind, and N, Tsiftes, 2009. Contiki IPSN 2015 Tutorial Handouts, Sics.Se, p. 6.
[13]. A., Aijaz, and A.H, Aghvami, Cognitive machine-to-machine communications for internet-of-things: A protocol stack perspective, IEEE Internet Things Journal., vol. 2, no. 2, pp. 103112, 2015.
[14]. C. Thomson, "Cooja Simulator Manual," no. C, pp. 2014.