Energy-aware Routing in Internet of Things (IoT) Networks

Shahzad Sarwar
Punjab University College of Information Technology (PUCIT)
University of the Punjab, Lahore, Paksitan
s.sarwar@pucit.edu.pk

Saamia Rauf
Punjab University College of Information Technology (PUCIT)
University of the Punjab, Lahore, Paksitan
mscsf16m025@pucit.edu.pk

Rashid Rasheed
Punjab University College of Information Technology (PUCIT)
University of the Punjab, Lahore, Paksitan
phdcsf15m506@pucit.edu.pk

Laeeq Aslam
Punjab University College of Information Technology (PUCIT)
University of the Punjab, Lahore, Paksitan
laeeq.aslam@pucit.edu.pk

***Abstract*— In contrast to wireless sensor networks (WSN), Internet of things (IoT) applications are supported by global IP connectivity and two-way data communication between low-power wireless sensor nodes of lossy area networks. The Internet Engineering Task Force (IETF) has proposed the routing protocol for low power and lossy area networks (RPL). The RPL constructs a destination oriented directed acyclic graph (DODAG) that is grounded at root node. The DODAG is constructed by using an objective function (OF) to compute rank of each sensor node. The IETF has proposed two objective functions that are OF0 and MRHOF. Subsequently, with the help of the rank of nodes, a parent-child relationship is established between nodes to construct the DODAG. In this paper, we propose an objective function 'EEQ' that protects the node which has already excessively consumed it's energy for forwarding sensed data towards the root node. The proposed objective function EEQ computes rank of a node using three energy related parameters i) expected number of transmissions, ii) consumed energy, and iii) queue length. Under low and high intensity traffic load, we have simulated various scenarios for OF0, MRHOF, and EEQ. The simulation results show that for high intensity traffic, EEQ has less overhead of control messages as compared to OF0 and MRHOF, resulting in energy conservation.**

Keywords—IoT, routing, RPL, objective function

I. Introduction

Conceptually, Internet of Things (IoT) is a framework of heterogeneous devices and their interconnections which creates a shared global information base. Alternatively, one may like to view IoT as a mechanism to access and control *things* using *wireless networks* over the *Internet*. These mechanisms have fueled development of wide range IoT applications from industrial control systems to smart cities and agriculture applications. Such applications benefit the society in terms of enhanced productivity, effective resource planning, and predictive maintenance of machinery and devices through real-time remote monitoring and controlling mechanisms. Conventionally, the three-tier architecture of IoT application is comprised of i) sensing and actuation, ii) network, and iii) application tier, as shown in Figure 1.

Application
Network
Sensing and Actuation

Fig. 1. Architecture of IoT application.

HTTP,FTP, SMTP, DNS	Application Layer	CoAP, XMPP, MQTT, AQMP
TCP/ UDP	Transport Layer	TCP/ UDP
OSPF, RIP, BGP IPv4/IPv6, ICMP	Network Layer	RPL IPv6, 6LoPWAN
Ethernet, WiMAX	Link Layer	802.15.4
Wired or wireless Medium	Physical Layer	Radio Transmission

Fig. 2. TCP/IP Stack versus IoT Stack

In a geographical area, sensor nodes make periodic observations for a physical phenomenon for example temperature, humidity, light, and air quality. Later, the sensed data is routed through the network in order to furnish the data requirements of an IoT application. However, the domain of IoT faces many challenges that need special attention of the research community.

For IoT applications, the most important goal is the longevity of the services. Which turns out to be a really challenging task in the presence of meager resources of wireless sensor nodes. In general, tiny battery-operated sensor nodes, come with low price-tag, have limited memory, computational, and communication capabilities. Furthermore, we cannot afford liberal usage of computational and communication capabilities more specifically on account of scarce battery power. Otherwise, the network nodes would quickly become dead that would resultantly disrupt the network services required by IoT applications. Similarly, it is evident that running full blown fancy Internet protocols over lean nodes is simply an inappropriate choice. To address this issue, bare-bone lightweight protocols have been proposed by the Internet Engineering Task Force (IETF). The lightweight protocols are designed while keeping the header size and overhead of control messages to the minimum that reduces the energy consumption and results in enhanced longevity of the network.

The TCP/IP and IoT protocol stacks have been shown in the Figure 2. For IoT like networks, several lightweight application protocols have been proposed such as Extensible Messaging and Presence Protocol (XMPP), Message Queuing Telemetry Transport (MQTT), Advanced Message Queuing Protocol (AMQP), and Constrained Application Protocol (CoAP). The CoAP is the counterpart of Hypertext Transfer Protocol (HTTP) in IoT. Further, CoAP runs over User Datagram Protocol (UDP). However, CoAP can emulate Transmission Control Protocol (TCP) like behavior using its confirmable message(s) to meet the reliability requirement of an application [1]. At the network layer, IPv6 offers a wide range of IP addresses to the ever-increasing count of IoT devices. It has been estimated that in year 2020 some twenty billion devices would be connected to the Internet. But, for global connectivity, the processing of IPv6 40-byte long header would be a huge overhead for the battery powered IoT devices. For this, an adaptation layer named 6LoWPAN has been designed that defines header reduction scheme for IPv6 and UDP headers [2], to be used inside the IoT networks. The 6LoWPAN adaptation layer provides a mechanism for mapping 40-bytes IPv6 header to 4-byte RPL headers and vice versa. The IETF working group for 6LoWPAN is currently working on TCP header encoding but they have not standardized such encoding schemes yet [3, 4]. Further, the role of 6LoWPAN is to fragment and reassemble IPv6 packets and provide an encapsulation to IPv6 packets into 802.15.4 frames at MAC level [5]. The conventional complex routing protocols require large memory and computational resources. But, low power lossy area network (LLN) are unable to support legacy routing protocols. The IETF working group for routing over low power and lossy area networks (ROLL) has proposed a routing protocol for LLN (RPL) [6]. In WSN, we have several MAC layer protocols such as WiFi, Ultra-Wide Band (UWB), Bluetooth, and 802.15.4 standard. For IoT, the Internet Engineering Task Force (IETF) has chosen 802.15.4 standard at physical layer and for MAC.

It is evident that routing protocol RPL plays pivotal role in forwarding crucial data for the IoT applications. RPL is a gradient routing scheme, in which participating nodes are ranked according to certain objective function. The rank of a node is used to select the preferred parent by the child node in an attempt to construct the directed acyclic graph (DAG) with a particular root node. There are two proposed objective functions in the literature that are OF0 and minimum rank with hysteresis objective function (MRHOF). The rank of a node in OF0 represents number of hops to the root node. The MRHOF uses the number of expected transmissions (ETX) for a successful packet transmission to the root node as a link quality measurement to assign the rank of a node. Notably, these OFs are deficient to cater the node characteristics such as energy consumed and queue length at a senor node. Thus, the resultant DAG would be sub-optimal in terms of load balancing, energy conservation, and network lifespan. In this paper, we proposed an objective function that takes into account ETX, energy consumed, and queue length, as well.

Section II presents an introduction to routing protocol for low power lossy network (RPL) and routing in IoT. In Section III, the related work of objective function is presented. Section IV explains proposed objective function and Section V describes simulation setup. Results have been presented in Section VI and finally, Section VII presents conclusions and future work.

II. Routing Protocol for low power lossy network (RPL)

RPL is used at network layer for data routing in low power lossy network (LLN). RPL is a distance vector protocol which creates a Destination Oriented Directed Acyclic Graph (DODAG). The distance vector protocol does not require any knowledge of network topology in advance. A wireless sensor node can become the part of existing network by passing the control messages. The purpose of distance vector is to find the minimum distance between the nodes by maintaining a vector table and communicating it with other nodes [7]. Figure 3a is showing few nodes before making a network. In order to communicate with each other these nodes need to connect with each other using some common mechanism. While getting connected in a network there is a need to avoid a loop or cycle [8], so that messages always reach to their destination. This problem can be resolved by making directed acyclic graph (DAG), as shown in Figure 3b. Also, RPL creates a destination oriented directed acyclic graph (DODAG) that points towards a root node, as shown in Figure 3c. The root node is usually a live powered node which acts as a gateway between an IoT network and the Internet. This root node gathers the data from the nodes in its region, aggregate it, and sent it over the Internet to the cloud data centers and vice versa.

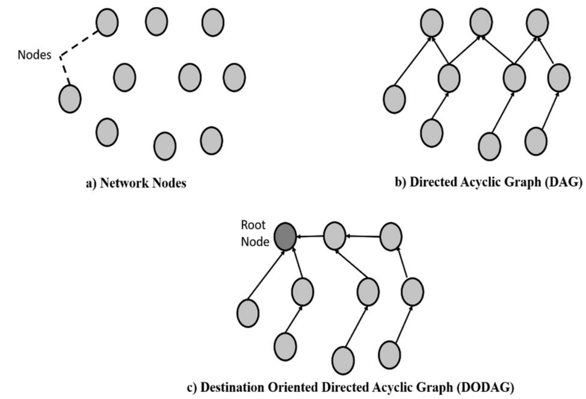

Fig. 3. RPL Network Graph

A objective function is used to calculate the rank of a node with in a DODAG. Rank defines the position of a node relative to the other nodes. A disconnected node has an infinity rank until it becomes the part of a DODAG after receiving some rank. In a DODAG, rank monotonically increases towards the leaf nodes as we move from root node , as shown in Figure 4a. A node chose its preferred parent node that has the least rank among potential parent nodes, as shown in Figure 4b.

DODAGs are further categorized into two types: grounded DODAGs and floating DODAGs. When every node in a DODAG is connected to the root it is called a grounded DODAG. The floating DODAG is a disjoint part of a DODAG which is not providing connectivity to its child nodes to the root node. Floating DODAG can occur during the RPL repair operation. RPL has two types of modes namely, storing mode and non-storing mode. In storing mode each node maintains a routing table of downwards traffic path. Packets travel only to a common parent. Nodes having lower ranks may have bigger routing tables. Due to this reason the storing mode is limited by the size of routing table. Otherwise, memory overflow will occur and protocol may fail. While in non-storing mode root node has the routing

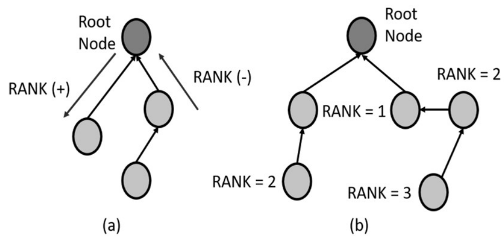

Fig. 4. DODAG Nodes Rank

table and all the data traffic is sent to the root node that further redirects it towards the respective node.

RPL can support three types of communication scenarios: (i) point-to-multipoint (P2MP), (ii) multipoint-to-point (MP2P), and (iii) point-to-point (P2P). RPL always keeps control information messages separate form data messages.

RPL uses Internet control message protocol (ICMP) for sending control information. In RPL, there are total four type of control messages namely i) DODAG Information Solicitation (DIS), ii) DODAG Information Object (DIO), iii) Destination Advertisement Object (DAO), and iv) Destination Advertisement Object Acknowledgement (DAO-ACK). Base field of ICMP message contains these control messages in the form of objects. The options field may contain optional data. Metric container that is present in a DIO base object is used to report constraints along the DODAG. Based on these constraints a node selects its preferred parent in a DODAG and calculates its rank. Upon receiving a message, each node compares its rank with respect to its parent. Parent selection is based on objective function and hysteresis [9].

There are two types of constraints: node constraints and link constraints [6]. Node constraints can be node attributes like CPU requirement or memory usage, power consumption and hop count to the root node. While link constraints can be throughput, latency, and reliability.

Each node sends out DIO messages periodically according to a trickle timer to other nodes. The trickle algorithm has been developed to reduce the control messages overhead while constructing the DODAG topology. The suppression mechanism is enabled when a node realizes that its neighbors have received enough number of transmissions for the same piece of information, it suppresses the transmission of information for the current time interval of the trickle timer. But, inconsistent state resolution mechanism removes the inconsistency by sending control messages at higher rate than before [10, 11].

III. Objective Functions Related Work

In this section, we present various objective functions along with their merits and demerits, which have served as foundation to our motivation.

Before going into exercise to list and compare various objective functions, let us quickly summarize details that are pivotal in understanding the RPL. In the Section II, we have explained in detail that RPL describes the method to construct the DODAG as each node selects a preferred parent node with the lowest rank and joins the current optimized route to the grounded root node. Besides selection of a preferred parent each node maintains a list of successors for backup to be used in case the preferred parent fails. For each node, the rank is updated based on the rank of its preferred parent. Thus, rank of nodes monotonically increasing starting at the root node towards the leaf nodes. The rank is computed with the help of objective function (OF). An OF defines a criterion to compute the rank of a node, based upon one or more (node/link) metrics for example hop-count to the root node. In the literature, several OFs have been proposed and evaluated in terms of overhead, throughput, and convergence time.

For computing the rank of a node, default objective function (OF) of RPL is known as OF-Zero, the short from is OF0 [12].In this objective function, rank is computed on the basis of hop-count, only. The hop-count is considered as a node metric. However, in this objective function other node or link metrics can be used as well [12]. Intuitively speaking, it is expected that nodes with smaller rank would burn out earlier. Because, such nodes have to forward the data of children nodes, to and from the root node thus resulting in short lifespan of IoT network. We expect energy consumption at a node is a function of expected transmissions depending on the quality of wireless link.

Further, ROLL the working group of IETF has proposed Minimum Rank with Hysteresis Objective Function (MRHOF). In MRHOF, number of expected transmissions (ETX), is used to compute rank of a node. ETX is a metric to capture the link parameter that is quality of link. In low power lossy area network (LLN), link reliability can be measured with expected transmission count (ETX) which actually is an average number of transmissions a node required to successfully deliver a packet. Parent with the minimum path cost will be selected as preferred parent. Later on, candidate parent for this node cannot be selected until its rank is smaller, by a certain threshold per say 30%, than current parent. If a parent is switched and rank is updated with smaller value, it has to be communicated to the downstream nodes for updating their ranks. This hysteresis mechanism has been found effective for reducing the possible parent changes. Correspondingly, it decreases the overhead due to exchange of control messages to the downstream nodes that occurs when parent is change [13].

In [14], a hybrid OF (HOF) has been presented that includes two metrics i) hop-count and ii) number of expected transmissions (ETX). The weights of two metrics are assigned using a machine learning technique in order to compute the rank. However, a node switches its parent only if both metrics of candidate parent are smaller at least by a margin of the predefined thresholds than those of present parent. The simulation results have been presented and compared with two objective functions in terms of number of throughput, parent changes, control traffic, and energy consumed [14]. Furthermore, it has been observed that throughput is close to MRHOF for a tuple of weights.

Since, objective functions OF0 and MRHOF follow a greedy approach for selecting a parent. That may cause the problem of node overloading, of the one that has more children than other nodes, in an unbalanced DODAG of the network. Especially under excessive traffic load, nodes closer to the root nodes would start failing which would result in early expiry of the wireless network. Thus, it can be concluded, the DODAG construction of RPL is ignorant of load balancing aspect.

In order to address this issue a solution has been presented in [15]. A novel load balancing OF (LBOF) has been proposed in which rank represents number of children of a node, which is known as child node count (CNC). To fulfill this requirement, they have presented three new modifications in RPL. i) introduce new RPL Metric called child node count (CNC) object, ii) amend preferred parent IP address in DIO message by each node, iii) now parent node accepts a DIO from its child to keep track of its child nodes. Further, in the view of load balancing, a traffic aware objective function (TAOF) has been proposed in [16]. The TAOF uses packet transmission rate as a metric to decide the rank of node, which helps to achieve the load balancing among nodes of the network. This is a proposed solution for load balancing not yet implemented or tested. For this reason, it is difficult to comment on its effectiveness in terms of energy consumption.

In the literature, we can find the performance evaluation of RPL with respect to various load balancing objective functions (OFs) [15, 16]. However, it is evident that the DODAG construction, by using either of the OFs, does not take into account the two important aspects of nodal characteristics such as consumed energy and the queue length.

IV. Proposed Objective Function

In this paper, we present an object function (OF) named as EEQ. This OF takes three parameters as input to compute the rank of the node. The three parameters are i) number of expected transmissions (ETX), ii) energy consumed, and iii) queue length of the node. The values of three parameters would be normalized to 1 using the highest value of each parameter. Let us assume energy consumed and queue length is represented with *E* and *Q*, respectively. Using the objective function EEQ, the rank of a node can be obtained by the Equation 1.

$$Rank = \propto . ETX + \beta . E + \gamma . Q \qquad (1)$$

Here, α, β, and γ are the weights of parameters ETX, E, and Q, respectively. Further, the value of α, β, and γ ranges between 0.0 to 1.0 such that

$$\alpha + \beta + \gamma = 1.$$

Also, for this study, we assume equality among these weights, that means

$$\alpha = \beta = \gamma.$$

However, how to decide optimal weights according to the importance of parameters is beyond the scope of this study.

Table 1: SIMULATION PARAMETERS

Name	***Value***					
Area	100 X 100 meter					
Topology	Random					
Transmission range	25 meters					
Simulation time	1800 second					
Objective functions	OF0		MRHOF		EEQ	
Weights of EEQ metrics	$\propto$		β		γ	
	0.33		0.33		0.33	
Traffoc Intensity	No. of motes			Time interval of data sensing and subsequent transmission (sec)		
Low	40	60	80	300	600	900
High	10	20	30	10	30	50

Besides transmission quality, represented with (ETX), this EEQ objective function captures the node characteristics in terms of present traffic load, using queue length Q, and also tracks the past with the help of energy consumed E. The information of the present and the past would help us to decide the role of node in the future.

V. Simulation Setup

In this research, to simulate our proposed solution and existing RPL techniques we have used COOJA a Contiki based network simulator. Contiki is an open source operating system for IoT platforms. In our simulation, we have used unit disk graph medium (UGDM) with 1000 milliseconds mote startup delay. We have tested our proposed objective function on random topology of several nodes with low and high intensity of traffic intensity. Table 1 summarizes the simulation setup that we have used in our experiments.

The performance of RPL can be enhanced in terms of energy consumed by tweaking the thresholds of hysteresis [13] for parent change and the trickle algorithm. However, in this study, in order to make a fair comparison of various objective functions without diluting in this dimension, we restrict ourselves to the basic definitions of hysteresis and trickle algorithms.

VI. Results and Discussion

The overhead of control messages has been compared for OF0, MRHOF, and EQQ. The overhead of control messages is computed using the Equation 2.

$$\%\, Overhead = \left[\frac{Control\ Messages}{Control\ Messages + Data\ Packets}\right] * 100 \qquad (2)$$

For better interpretation of results, it must be noted that large interval of time between consecutive data sensing events would generate low traffic per unit of time. Also, along the x-axis of plots the traffic intensity is taken as inverse to data sensing time intervals in seconds.

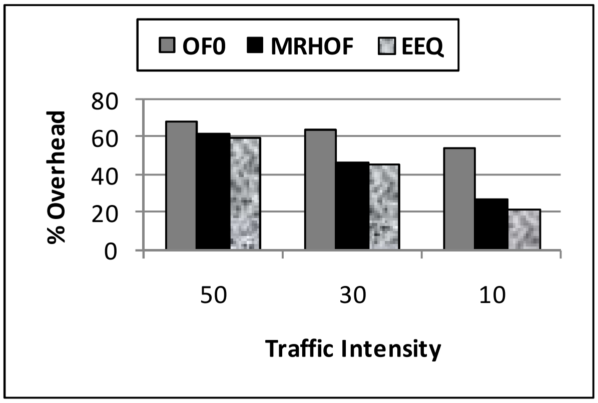

Fig. 5. Control messages overhead with 10 nodes.

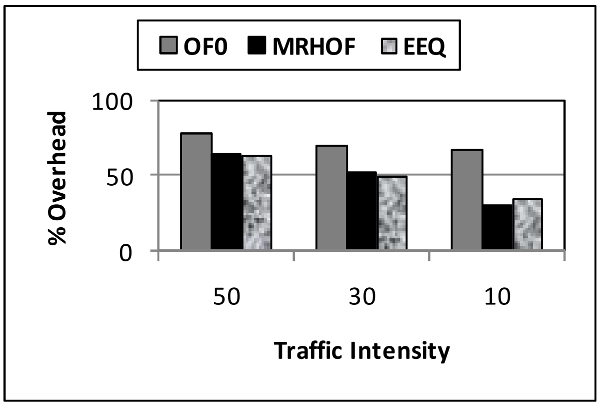

Fig. 6. Control messages overhead with 20 nodes.

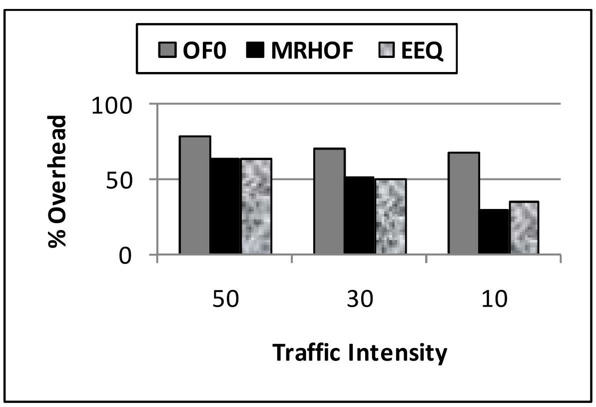

Fig. 7. Control messages overhead with 30 nodes.

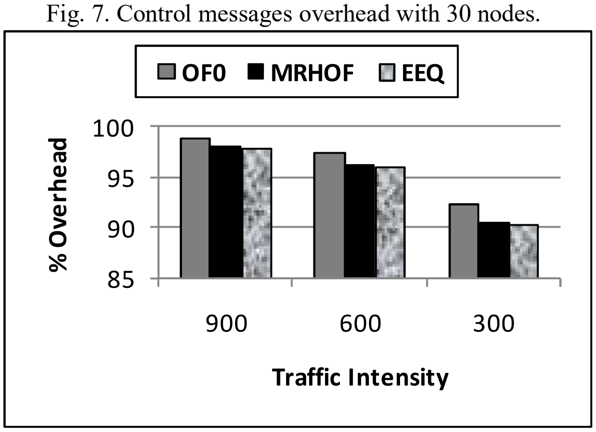

Fig. 8. Control messages overhead with 40 nodes.

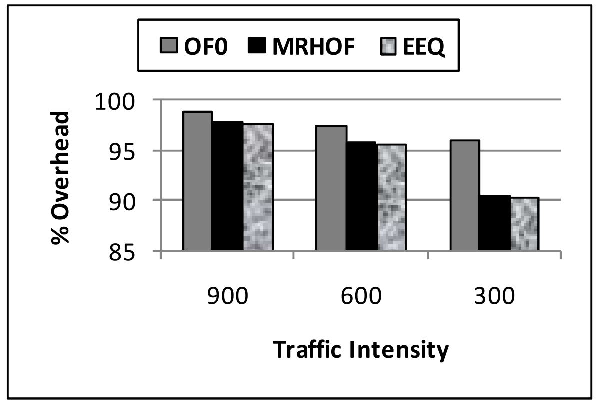

Fig. 9. Control messages overhead with 60 nodes.

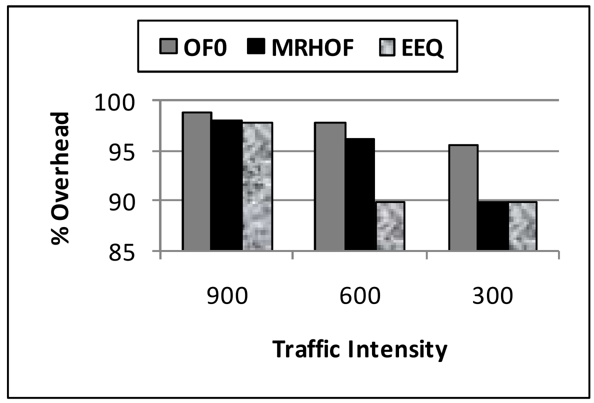

Fig. 10. Control messages overhead with 80 nodes.

For high intensity traffic, the percentage overhead of control messages of a network comprised of 10, 20, and 30 nodes are presented in Figures 5, 6, and 7, respectively. In the Figure 5, the results show that EEQ incurs 20% overhead of control traffic when the traffic intensity is high when data sensing is scheduled after every 10 sec. Also, it can be seen that performance of EEQ is better than OF0 and MRHOF.

For high intensity of traffic and sparse network of ten-nodes, the results show that our proposed objective function is performing better than OF0 and MRHOF objective functions in terms of control traffic overhead. Further, EQQ outperforms the OF0 with a significant margin in relatively more dense networks comprised of 20 and 30 nodes, as shown in Figures 6 and 7, respectively. Also, EEQ without hysteresis induces almost same amount of control messages as MRHOF.For low intensity of traffic and dense network comprised of 40, 60, and 80 nodes the results are presented in Figures 8, 9, and 10, respectively. Although, these results show that EEQ performs equal to MRHOF. Yet, EEQ is better than OF0 and incurs less overhead of control messages. It can be observed that the penalty of control messages is reaching up to 98% for all objective functions when the traffic intensity is low and network is densely populated. That means almost all energy of a node is being consumed just for processing the control traffic.

VII. Conclusions and Future Work

In this study, we conclude that objective function EEQ has proved it significance in high intensity of traffic, which is the scenario of upcoming IoT applications, in the near future. It has been identified that EEQ with hysteresis would be very good candidate to challenge the MRHOF. Further, employing an intelligent method to decide the optimal weights of multi-parameter objective functions, such as EEQ, can give enhanced performance. Furthermore, we propose to compare the performance of EEQ with other objective functions for example multi-parameter load-balancing objective functions in terms of parent changes, longevity of the network, throughput, and delay.

REFERENCES

[1] B. Leiba, et al., “The Constrained Application Protocol (CoAP)”, RFC 7252, 2014.

[2] T. Zheng, A. Ayadi, X. Jiang, "TCP over 6LoWPAN for Industrial Applications: An Experimental Study", 4th IFIP IntI. Conf New Technol. Mobility and Security (NTMS), pp. 1-4, Feb. 2011.

[3] A. Ayadi, et al.; INTERNET DRAFT; Expires April 28, 2011; TCP header compression for 6LoWPAN; October 25, 2010.

[4] C. Gomez, et al.; INTERNET DRAFT; Expires December 12, 2018; TCP Usage Guidance in the Internet of Things (IoT); June 10, 2018.

[5] G. Mulligan, et al., “The 6LoWPAN Architecture”, EmNets 2007, Cork, Ireland.

[6] A. Farrel, et al., “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks”, RFC 6550, 2012.

[7] D. Waitzman, et al.; RFC 1075; “Distance Vector Multicast Routing Protocol”; November 1988.

[8] P. Mannan, and T. Jayavignesh, “Alternate simplistic approach to solve count-to-infinity problem by introducing a new flag in the routing table”, 2016. Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Pages: 291 - 295, Kolkata, India.

[9] N. Accettura, L. Grieco, G. Boggia, and P. Camarda, "Performance Analysis of the RPL Routing Protocol", in IEEE International Conference on Mechatronics, Apr. 2011, Istanbul, Turkey.

[10] Ghaleb, A. Al-Dubai, E. Ekonomou, "E-Trickle: Enhanced Trickle Algorithm for Low-Power and Lossy Networks", 14th IEEE International Conference on Ubiquitous Computing and Communications (IUCC-2015), pp. 1123-1129, 2015.

[11] B. Djamaa, M. Richardson, "Optimizing the Trickle Algorithm", IEEE Communications Letters, vol. 19, no. 5, pp. 819-822, 2015.

[12] Thubert; RFC 6552; RPL Objective Function Zero; March 2012.

[13] Gnawali & Levis; RFC 6719; MRHOF; September 2012.

[14] Sheeraz A. Alvi, Fakhar ul Hassan, and Adnan Noor Mian; On the energy efficiency and stability of RPL routing protocol; 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC); Pages: 1927 - 1932; Valencia, Spain.

[15] Qasem, et al.; INTERNET DRAFT; Load Balancing OF for RPL; October 29, 2017.

[16] Ji, et al.; Expires June 9, 2018; Traffic-aware OF; December 2017.