RPL LOAD BALANCING IN INTERNET OF THINGS

Mohammad Reza Parsaei[*]**, Ahmad Reza Parnian,**
Samaneh Miri Rostami and Reza Javidan

IoT Laboratory, Department of Computer Engineering and Information Technology,
Shiraz University of Technology, Shiraz, Iran.

*Corresponding author: mr.parsaei@sutech.ac.ir

ABSTRACT: The wide address space provided by Internet Protocol version 6 (IPv6) lets anything be uniquely identified. Consistency of the modified version of the IPv6 protocol stack with smart objects, facilitated the Internet interconnection of smart object networks and introduced what is known as "Internet of things". A smart object is a small micro-electronic device that consists of a communication device, a small microprocessor, and a sensor or an actuator. A network made of such devices is called a low-power and lossy network. RPL routing protocol that is consistent to IPv6, is designed to be used in these kinds of networks. Load balancing is not considered in the RPL design process. Whenever RPL is used in large scale low-power and lossy networks, some nodes will suffer from congestion and this problem severely degrades network performance. In this paper, we consider solutions provided to tackle RPL load balancing problems. Load balancing algorithms and protocols are evaluated through simulation. We evaluate IETF RPL implementation and LB-RPL method with Contiki OS Java (COOJA) simulator. They are assessed comprehensively through metrics such as Packet Delivery Ratio, Average End-to-End delay, and Gateway Throughput. LB-RPL improves RPL in terms of Packet Delivery Ratio and throughput but increases Average End-to-End delay. Simulation results show that RPL load balancing needs extensive work to be performed.

ABSTRAK: Ruang alamat yang luas yang disediakan oleh Protokol Internet versi 6 (IPv6) memberikan identiti yang unik bagi setiap objek untuk dikenal pasti, selaras dengan versi yang telah diubah oleh timbunan protokol IPv6 melalui objek pintar, ini membantu sambungan Internet dalam rangkaian dan memperkenalkan perkara-perkara Internet (Internet of Things). Objek pintar adalah komponen mikro-elektronik yang kecil yang terdiri daripada peranti komunikasi, mikro-pemproses kecil dan sensor ataupun pemproses signal (actuator). Rangkaian yang terdiri daripada komponen-komponen ini disebut rangkaian rendah ataupun kurang tenaga. Protokol penghalaan RPL yang konsisten dengan IPv6, telah direka untuk digunakan dalam rangkaian ini. Walau bagaimanapun, beban sama rata tidak dikira dalam rekaan proses RPL ini. Apabila RPL digunakan dalam rangkaian sistem yang luas, kurang ataupun rendah tenaga, sebilangan nod akan terlibat dengan kesesakan dalam rangkaian dan masalah ini menyumbang kepada prestasi rangkaian akan terjejas teruk. Kertas ini mengambil kira solusi yang disediakan untuk mengatasi masalah keseimbangan beban RPL. Algorithma keseimbangan beban dan protokol telah dinilai melalui simulasi. Kerja ini juga mengambil kira implementasi ke atas IETF RPL dan kaedah LB-RPL dengan mengunakan simulasi Contiki OS Java (COOJA). Ianya dinilai secara menyeluruh dengan menggunakan matriks seperti Nisbah Paket Dihantar (Packet Delivery Ratio), purata kelewatan nod hujung-ke-hujung (average end-to-end delay), dan laluan masuk (gateway throughput). LB-RPL dapat menambah baik RPL dari segi Nisbah Paket yang dihantar dan laluan transmisi, tetapi menambahkan purata kelewatan transmisi data

hujung-ke-hujung nod. Hasil simulasi menunjukkan beban keseimbangan RPL memerlukan kerja pembaikan yang banyak pada masa akan datang.

1. INTRODUCTION

The digital revolution in the twenty-first century is more impressive than the past digital revolution. During the twentieth century, the world experienced two wonderful and important digital revolutions: first, computers found their way into homes and offices, and then the Internet connected computers to each other, fundamentally changing the interaction between people and the digital world.

Now in the twenty-first-century revolution (smart objects), Internet of Things (IoT) links the digital world to the physical world. Industry forecasts the number of intelligent objects over the next ten years could be on the scale of billions. In the next decade, the world will see that IoT will change human behaviour in regard to both the physical and digital world [1-4].

With the increasing development of microcontrollers and microelectronic technologies, smart objects have been introduced. A smart object is a small microelectronic device that includes a communication element (typically low-power RF), a small microprocessor, and a sensor or actuator. A large number of smart objects used in various applications establish networks of intelligent objects. These kinds of networks are called low-power and lossy networks due to the specific features of the smart objects. They use low-power communication radios with limited transmission rate. IEEE 802.15.4 technology is the most common communication technology in such networks [5]. IPv6 protocol is selected as the best option for Internet of Things because it supports a very large address space. Some efforts were done by IETF in order to define a standard for this [6, 7]. The 6LOWPAN group was formed to standardize the adaptation of IPv6 in IEEE 802.15.4 based networks. They were ordered to determine how to transmit IPv6 packets on IEEE 802.15.4 communication links and how to implement and configure IPv6-based sub-networks [8]. Another group was established to standardize IPv6-based routing protocol independent of the communication link of equipment with limited resources [9-11]. In recent years, several routing protocols have been suggested for low-power and lossy networks. The best one is RPL, which was standardized by the IETF in 2012. RPL technology is used for a wide range of LLN applications such as building automation, urban sensor networks [13], and smart grid communication [6].

2. INTERNET OF THINGS (IoT)

Internet of Things is the greatest challenge and opportunity for today's embedded IP-based devices such as sensors, home appliances, machinery, building automation equipment, and even toys. Over the past decade, advances achieved in terms of microcontrollers, power, and microelectronics technology motivated industry to benefit IP-based smart devices (called Smart Objects) services on the Internet. These services not only contain data created by humans but also data highly related to the physical world including sensor data, monitoring and control machines, and other types of physical items. This new field of Internet provides applications that are crucially related to stability, efficiency, and safety. Building automation, health, energy efficiency, smart grid communication, environmental monitoring are instances of such applications [1-3].

Exact estimation of IoT dimensions is difficult. Users don't affect its growth. Due to a rapid increase in the number of embedded IP-based devices, it is assumed that soon the IoT will outpace all other equipment on the Internet in terms of size (number of nodes) and that its growth will continue to increase at a high rate. The greatest potential for future growth is low-power wireless networks and equipment that are not yet IP-based. 6LoWPAN technology was developed to realize Internet of embedded devices. This will be achieved by simplifying IPv6 capabilities and header compression format and by taking the nature of wireless networks into account [14].

3. LOW-POWER AND LOSSY NETWORKS

A typical smart object has several kilobytes of memory, a small microcontroller, and a limited power supply. Smart objects can form sensor networks that potentially consist of hundreds of thousands of nodes. Such networks are commonly known as low-power and lossy networks (LLN). The name derives from serious energy restrictions and deployment of these networks in unpredictable environments. Low-power and lossy networks equipment have limited resources such as the processor, memory, battery, and unstable radio communications. LLN equipment pieces are connected to each other with a variety of communication links such as Bluetooth, WiFi, low power IEEE 802.15.4, or PLC (power line communications). Using IEEE 802.15.4 is significantly pervasive than other technologies [14].

LLN networks can also be used for physical monitoring in various environmental conditions. Some examples of these networks are environmental monitoring, home automation systems remote control, patient's vital signs monitoring, and intelligent building wireless lighting control. Usual monitoring parameters are temperature, humidity, pressure, voltage in power lines, and vital signs. In these cases, the equipment pieces are sensors that consist of a microcontroller, transmitter, receiver, and power supply. Sensors are very small in size. They are powered by batteries, solar cells, or electricity, so energy efficiency is essential. LLNs are applicable to low traffic networks. LLN equipment uses short-range radio media, so it is prone to error bit. Lossy communication is characteristic of LLNs. According to the vastness of LLN topology, new routing protocols and mechanisms that compromise with their specific features must be served. LLN Lossy nature has a significant impact on the design of their routing protocols. Since the link failures are frequent and usually transient, the routing protocol must not show a serious reaction to the network convergence because it is a temporary downtime. Due to the mentioned reasons, finding the best path for data delivery is one of the challenging issues in LLNs, so an efficient routing mechanism should be used to find and keep track of the changes in the network path. The routing mechanism should consider both natures of LLNs i.e. resource constraints and the lossy media [15].

4. ADAPTATION LAYER 6LoWPAN

The IETF 6LoWPAN working group [6LoWPAN] have been established to deal with problems in order to make possible usage of IPv6 in LLN networks. Special features of IPv6, such as simple header structure and hierarchical addressing scheme, make it ideal for use in 6LoWPAN-based LLN networks. Also, organizing a dedicated group of standards for such networks provides the minimum requirements for the implementation of IPv6 with a light 6LoWPAN stack that can be used for most limited-resource equipment. Eventually, the specific characteristics of LLN have been considered by designing a new version of neighbour discovery protocol for 6LoWPAN. So, 6LoWPAN is an efficient

development of IPv6 for LLN networks that provides end-to-end IP-based communication in a wide range of embedded applications [1, 3].

In this paper, RPL routing protocol and its load balance problem are introduced. Several suggested versions of RPL routing protocol designed to tackle load imbalance problem are also considered. LB-RPL is the most comprehensive among all the versions. RPL and LB-RPL are evaluated in a simulation scenario. Section 5 introduces RPL and presents related works performed about RPL load balance. RPL and LB-RPL routing mechanisms are explained in Section 6. Section 7 presents simulation results and Section 8 concludes the paper.

5. RELATED WORKS

5.1 RPL Routing Protocol

IETF assigned a ROLL working group to determine the appropriate routing protocol for low-power and lossy networks according to the documented requirements with headings of routing requirements for home, building, industry and urban automation [16-19]. Major mentioned requirements in this documentation are as follows:

- Route tracking: information such as path quality, the number of nodes, and the replaced active route with its associated costs should be available to facilitate route discovery. The path quality is a relative metric determining priority of the chosen path from the source node to the destination node in comparison to other replaceable routes. The metric may be a function of the number of hops, signal strength, active routes or any other criteria.

- Route selection: reliable route selection and high quality communication link determination needs to be possible since the problem of joining nodes in the network initial setup is mitigated over time.

- Route cost: Routing protocols must support path quality criteria. The criteria include signal strength, available bandwidth, number of hops, amount of available energy and communication error rate.

- Path flexibility: paths need to be flexible and should improve selection criteria (e.g. signal quality and the number of visited hops) during the converging time.

- Alternate routes: the network layer should determine and establish second and third alternate routes to be used in failure time.

- Constraint based routing: some nodes may lose their battery power faster than other nodes or some others may have a limited energy source. Some nodes have a larger memory space, so they are able to save more neighbourhood information. Some others have a powerful processor and hence they can collect data in more sophisticated ways. In such situations, routing protocol must also support constraint based routing. Constraints such as the processor, memory size, battery level, etc. In other words, the routing protocol must be able to advertise the node's capabilities.

ROLL organized a detailed analysis on evaluating existing routing protocols, such as OSPF [20], IS-IS [21] and OLSR [22] over a set of applications broad enough to encompass common uses for LLNs. ROLL designed Routing Protocol for Low Power and Lossy Networks (RPL), since it found existing routing protocols, perform unsatisfactorily in their current form. RPL addresses the core requirements [16-19] of LLN applications.

The ROLL provides an evaluation of existing routing protocols in Table 1. The symbol "?" indicates that a specified result has not been received.

Table 1: An evaluation of existing routing protocols with LLN criteria

Protocol	Table	Loss	Control	Link Cost	Node Cost
OSPF/IS-IS	fail	Fail	fail	pass	fail
OLSRv2	fail	Fail	?	pass	Pass
TBRPF	fail	pass	fail	pass	?
RIP	Pass	Fail	Pass	?	fail
AODV	Pass	Fail	Pass	Fail	fail
DYMO[-low]	pass	Fail	Pass	?	fail
DSR	fail	pass	pass	fail	fail

5.2 RPL Routing Protocol in LLN Networks

As far as possible, RPL routing protocol was designed in the modular form based on a large set of requirements listed in different documentation. RPL has been designed for LLN networks that include resource-constrained equipment communicating through lossy wired and wireless communication links. Many decisions taken in RPL design process are based on the specific characteristics of LLN networks.

The lossy nature of the LLN links is not the only LLN characteristic that affects RPL design process. As resources are restricted, control traffic should be reduced. In these networks, data traffic is usually limited and it is necessary to reduce control traffic as much as possible to save bandwidth and energy.

5.3 RPL Routing Method

RPL is an Ipv6 based routing protocol used in LLN networks. This routing protocol makes a Destination Oriented Directed Acyclic Graph (DODAG) of network nodes. Nodes organize vectors of DODAG and communication links among nodes organize DODAG edges. Nodes forward received and generated data traffic to their parents in order to reach data to DODAG root. LLN Border Router (LBR) acts as DODAG root and starts DODAG construction process by broadcasting DODAG Information Object (DIO) messages. When other nodes receive DIO messages for the first time, they join the DODAG by selecting the DIO sender as their parent and then broadcast DIO messages. This process continues and nodes join the DODAG.

5.4 Load Balancing

In the RPL routing protocol design process, load balancing and congestion avoidance are not considered. The traffic passing through parent nodes and the size of their sub trees were not considered in the parent selection process. This causes an unbalanced tree. There have been extensive studies on RPL routing load balancing. Some of them are in the following review.

5.4.1 Imbalanced Tree Algorithm

Tripathi offered a greedy algorithm in order to solve load balancing problem. He calculated the load imbalance factor for each level of the routing. In this way, nodes that are prone to congestion are identified. This method aims to balance the routing tree and minimize the load imbalance factor. The algorithm selects a parent for a node from three nominated parents by itself. The root node executes the algorithm and tends to select

parents which minimizes load imbalance factor. This is done periodically, keeping the tree as balanced as possible during the network lifetime. This greedy algorithm aims to keep the load among same-level nodes balanced. The simulation results show that this algorithm significantly increases the average packet delivery ratio and network lifetime. The main algorithm characteristic is that it only needs a partial knowledge of the network [23].

The proposed algorithm by Tripathi is only efficient for networks in which nodes generate the same amount of traffic. This method is completely centralized and is implemented by the root node. In addition, each period has significant delays. The difficulty of the balanced problem is NP [24-26] and the complexity of the proposed algorithm is O (N^2).

The proposed algorithm involves large computational complexity, so resources and time are highly consumed. The balanced tree construction is a complex mathematic problem. It needs a significant number of control messages. According to the specific conditions of low-power and lossy networks, especially poor communication links, the topology of these networks are completely dynamic and change constantly. Balancing such topologies needs a lot of effort to be repeated frequently.

5.4.2 TREEB Algorithm

Kulkarni proposed a method for load balancing. In this method, DODAG root knows the number of nodes in each subtree. Each node that wants to join the DODAG gets aware of the DODAG's node count and joins a DODAG with the lowest number of nodes [27].

In this method, nodes that want to join a DODAG can be aware of the DODAG size. This method has no effect on load balancing of each tree individually and just tries to keep the size of the trees the same. For example, this method can create trees with the same size and completely unbalanced. When there is only one root, this algorithm is completely ineffective and only increases overhead and slows tree creation process.

6. DESCRIPTION AND IMPLEMENTATION OF THE LB-RPL AND RPL PROTOCOLS

6.1 RPL Routing Protocol

RPL is an IPv6-based distance vector routing protocol utilized in LLN. This routing protocol organizes a DODAG of nodes in order to determine routing paths. It is possible to have multiple DODAGs when there are more than one LBR. RPL uses the DODAG ID to identify DODAGs. Tree kinds of ICMPv6 control messages are used in DODAG construction and maintenance. DIO is used in order to advertise DODAG and route construction. DODAG Information Solicitation (DIS) is propagated by nodes that are eager to receive DIOs and finally, the Destination Advertisement Object (DAO) is transmitted from higher level nodes toward the DODAG root in order to introduce routes toward them. RPL is a flexible routing protocol capable of serving in various application areas. Objective Function (OF) is what enables RPL to be customizable according to different applications. OF is defined through an equation containing metrics and constraints related to nodes and communication links. Application requirements define constraints and metrics. RANK is the computed value of OF. Every node has a RANK and it is calculated using the OF. When a node wants to calculate its rank, it first computes the OF result and then adds it to its parent RANK. So a node RANK is the summation of its parent rank and the value provided by OF. It is obvious that Rank value increases from

root toward leaf nodes. Rank is an indicator of the quality of the route that a node provides toward the root. In RPL, it is preferable to select a node that provides the lowest Rank as parent.

LBR starts the DODAG construction process by broadcasting DIO control packets. Receiver nodes extract OF and Rank from DIO and then compute their rank and select a DIO sender as their parent. After joining the DODAG, nodes can produce DIO messages and propagate them in order to advertise the DODAG. Receiving DIO, joining a DODAG and advertising DODAGs continues until all nodes join the DODAG. The DODAG member nodes receive different DIOs. In this case, they extract information from the DIO and calculate Rank based on the received DIO. If the DIO advertiser node provides a better path than the current parent toward the root, it will be selected as the parent. If not, it will be held in a preferred parent set by the DIO receiver node. Each holds five best alternate neighbour nodes in its preferred parent set.

6.2 LB-RPL Routing Protocol

Liu proposed a LB-RPL routing protocol that detects load imbalance in a distributed and non-intrusive manner. It optimizes the data forwarding path by considering both workload distribution and link-layer communication qualities. In this method, each node delays DIO message transmission based on the efficiency of its buffer. Each node saves DIO messages based on a priority of their received time and then sends its traffic data through a number of its parents which have sent DIO messages earlier.

LBR-RPL does not select a parent for a node as its primary parent, but it uses the workload differences. The data traffic is distributed among multiple parent nodes. This method modifies the DODAG construction procedure by incorporating two techniques. The first one is workload imbalance detection and signaling, and the other is load balanced data forwarding [28].

Each node delays DIO message transmission according to its buffer utilization in order to inform others about its workload traffic. In fact, a node sets a timer based on the efficiency of its buffer in the previous period and transmits the DIO message after the timer expiration. Determining an appropriate value for the timer is crucial for detecting load imbalance. Buffer capacity limitation significantly affects the data delivery ratio. Therefore, buffer utilization is used to quantify a node's workload. An appropriate value for the timer is set as following:

$$Ti = T0 * \text{Buffer Utilization Counter [28]} \qquad (1)$$

Where T0 is a constant value and Buffer Utilization Counter indicates efficiency of the buffer.

After a node receives multiple copies of the DIO message from different lower ranked nodes, it will form its parent set. The priority order of these parents is determined based on the time their DIO messages have been received. A parent will have a higher priority if its DIO has reached earlier than others.

6.3 LB-RPL Algorithm

Unlike RPL that selects a single node as parent from parent set, LB-RPL uses the top *k* parent nodes as the next hop for data forwarding. The data packet transmission probability by node *i* to node *j* as the next hop is calculated through:

$$f_{ij} = \frac{\left(1 - p_{ij}^{\ c}\right)}{\sum_{j=1}^{k}\left(1 - p_{ij}^{\ c}\right)} \tag{2}$$

Where P_{ij} is drop probability of the transmitted packet from node *i* to node *j* according to the node link state, and f_{ij} is the probability of packet transmission from node *i* to node *j*.

Packets being ready for transmission by a node are distributed among *k* parent nodes according to their link states. Moreover, when one of the parent nodes suffers from heavy workload in the current period, it will add more delay in the DIO message transmission in the next round. A parent node will be out of the top *k* preferable next hop in its children's parent table if it has a long delay due to heavy workload. This node will not be used as the next hop by its children for data forwarding, therefore workload imbalance can be alleviated. Pseudo-codes of LB-RPL algorithm [28] are shown in Fig. 1 and Fig. 2.

Algorithm 1: Sensor Node Initialization Procedure

1:     **Initialize** parent set and buffer utilization counter

2:     **Update** the latest received version number

3:     **Insert** the DIO message source into parent set according to the message arrival time

4:     **Calculate** its own rank value

5:     **Set** timer value *Ti*

6:     **Generate** a DIO message with its own rank number and the latest version number

7:     When timer *Ti* expires, **broadcast** a DIO packet with current rank and version number

---

Fig. 1: Node initialization procedure

Figure 3 is an example with *k*=2. A fraction of $\frac{p_{63}}{p_{63} + p_{64}}$ packets are sent to node S_3 and remaining packets are forwarded to S_4.

6.4 Evaluation of Proposed Approach

In the LB-RPL method, the concept of objective function and RANK are completely unused. Rank is totally ineffective in parent selection and path criteria and restrictions are not considered, so the quality of the path provided by a parent node cannot be measurable. DIO message arrival time can be result of collision, interference, lack of access to the channel and some other factors, so it is not necessarily due to a crowded buffer. Moreover, if the DODAG size is great, a considerable delay will be added to the network due to delaying DIO message transmission.

The purpose of load balancing is to make equal traffic for all nodes at the same level. Since in this method traffic will be sent by k nodes, there will be idle nodes in each period and if there is high-volume traffic, all k nodes will be congested in each period.

Algorithm 2: LB-RPL: Load Balanced RPL Routing Protocol

1:     A sensor node listens to the radio channel

2:     Once a message M arrives, check the type of the message

3:     **if** M is a DIO message **then**

4:        **if** New version of DIO **then**

5:           Invoke Sensor Node Initialization Procedure

6:        **else**

7:           **if** Current DIO version **then**

8:              **if** Rank value carried in the message is less **than** current node's rank **then**

9:                 Insert the DIO message source to parent set according to message arrival time

10:              **end if**

11:           **else**

12:              Discard this message

13:           **end if**

14:        **end if**

15:     **else**

16:        **if** M is a DAO message **then**

17:           Process it according to RPL

18:        **else**

19:        **if** M is a data message **then**

20:           **if** the data message comes from a child of current node **then**

21:              Increase workload counter

22:           **end if**

23:           Forward this message by choosing the first two parent nodes from parent table, and selecting one as next hop with probability

24:           **end if**

25:        **end if**

26:     **end if**

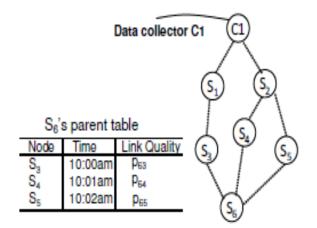Fig. 2: LB-RPL load balanced routing protocol.

Fig. 3: Message reception with time priority [28].

7. SIMULATION RESULTS

In order to consider LB-RPL in an exact manner, this routing protocol is simulated by Contiki COOJA simulator. Contiki was the first operating system that provided IP communication and is implemented in C programing language. The COOJA simulator is a Java-based sensor network simulator [29, 30].

In the simulation, a total number of 250 static nodes are randomly deployed in a 500 by 500 square meters area and one LBR is located at the top. Simulation is run for 3 hours or 10800 seconds. In order to obtain more accurate results, nodes begin to send UDP packets after 7 minutes of starting time. In the simulation scenario, all nodes send packets with a length of a hundred bytes to the root node. Each node sends two packets with an interval of 30 seconds per minute. Each node has a buffer which is enable to hold maximum 8 UDP packets. In order to get more accurate results, the simulation scenario is repeated 5 times with different random seeds. We evaluated LB-RPL through packet delivery ratio, AVG end-to-end delay and root node throughput. Table 2 summarizes simulation parameters.

Table 2: Simulation parameters

Parameter	Value
Simulation time	10800 seconds
Number of nodes	250
Traffic Type	UDP
Packet length	100 bytes
Area	500*500 m^2
Simulator	Contiki COOJA
Repetition	5 times
Buffer size	8 UDP packets

Figure 4 illustrates packet delivery ratio versus simulation time. This figure proves that LB-RPL performs better than RPL in terms of packet delivery ratio with a 90% confidence. The simulated network consists of a great number of nodes and congestion occurrence is completely probable in such networks. RPL is not optimized for large scale networks and cannot balance network load in order to prevent performance reduction. This leads to noticeable packet loss. LB_RPL performs better in this case since this protocol distributes load and directs data traffic among multiple paths. Therefore LB-RPL provides higher PDR.

Figure 5 illustrates AVG end-to-end delay versus simulation time. RPL performs better than LB-RPL in terms of AVG end-to-end delay and it is proven with 90% confidence. RPL makes routes based on the metrics and constraints formulated in OF. The OF used in RPL routing protocols is defined based on communication link quality. This results in better Rank for nodes providing a route toward the root with lower time. RPL provides better AVG end-to-end delay. In LB-RPL packets are distributed through the network, which results in longer paths. Packets may travel many nodes and links that are not completely in an optimized path toward the root. As LB-RPL does not use OF, it cannot formulate a delay in the routing process in order to decrease packets arrival time.

Figure 6 depicts root node throughput versus time. It can be seen that LB-RPL performs better than RPL with 90% confidence. LB-RPL prevents from intermediate nodes buffer over-utilization and lets packets to reach to destination through alternate routes even though they are longer. This decreased packet loss since buffer overflow is prevented. RPL selects routes with better overall Rank resulting in congestion in a part of network.
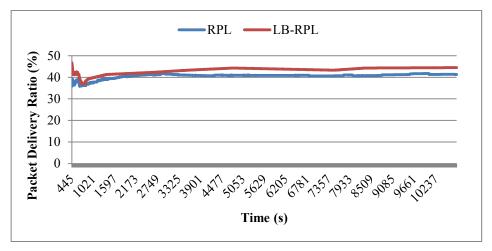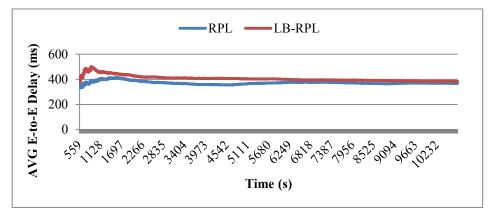

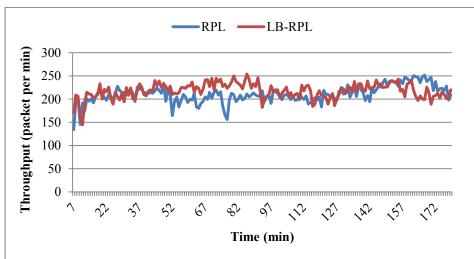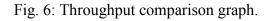Fig. 4: Packet delivery ratio comparing graph.

Fig. 5 Average End to End Delay comparing graph.

Fig. 6: Throughput comparison graph.

Simulation result analysis was performed according to [31, 32]. Initial data deletion was performed and the replication process through techniques presented in Tables 3 to 5 present the simulation result report and confidence intervals.

Table 3: The mean performance of the approaches for packet delivery ratio

Method	Mean	CI
RPL	41.005	(39.052 , 42.958)
LB-RPL	43.432	(41.272 , 45.592)

Table 4: The mean performance of the approaches for average end to end delay

Method	Mean	CI
RPL	372.532	(360.978 , 384.086)
LB-RPL	398.484	(390.507 , 406.461)

Table 5: The mean performance of the approaches for throughput of root node

Method	Mean	CI
RPL	210.543	(204.403 , 216.683)
LB-RPL	217.751	(212.196 223.306)

8. CONCLUSION

RPL routing protocol is the best choice for routing in low-power and lossy networks but Load balancing is not considered in RPL routing protocol standardization process by IETF. As RPL routing protocol and LLNs are used in large scale networks, Load balancing is an important issue to be considered.

Several methods have been proposed to alleviate the problem of load balancing. But they did not consider all aspects of RPL or some of them were not computationally efficient. LB-RPL is a new modification of RPL routing protocol, designed in order to tackle RPL load balance problems. This routing protocol distributes data traffic through the network and performs better than RPL in congestion prevention.

In this paper, RPL and LB-RPL were evaluated in terms of packet delivery ratio, AVG end-to-end delay and network throughput. LB-RPL performs better in terms of packet delivery ratio and network throughput. Although LB-RPL prevents congestion, it is not the mature solution for RPL load balance problem cannot formulate application routing requirements and does not support advantages that IETF provided in RPL design process. A new version of RPL that saves IETF standardized design specifications and balances load especially in large scale networks is completely required.

REFERENCES

[1]   Mazinani SM, Chitizadeh J, Yaghmaee MH, Honary MT, Tashtarian F. (2010) New Clustering Schemes for Wireless Sensor Networks. IIUM Engineering Journal, 11(1):51-69.

[2]   Arshad A, Khan S, Alam AZ, Ahmad FI, Tasnim R. (2014) A Study on Health Monitoring System: Recent Advancements. IIUM Engineering Journal, 15(2):87-99.

[3]   Parsaei MR, Parnian AR. (2015) IPv6 based routing in building automation network. In 2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI):1025-1031. DOI**:** 10.1109/KBEI.2015.7436185.

[4]   Parnian AR, Parsaei MR, Javidan R, Mohammadi R. (2017) Smart Objects Presence Facilitation in the Internet of Things. International Journal of Computer Applications, 168(4):25-31. DOI: 10.5120/ijca2017914345.

[5]   Adams JT. (2006) an introduction to IEEE STD 802.15.4. IEEE Aerospace Conference Proceedings: 1-8.

[6]   Deering S, Hinden R. (1998) Internet protocol, version 6 (IPv6) specification.

[7]   Ismail Y, Sarif SM. (2012) Availability of literature on engineering ethics in the internet. IIUM Engineering Journal, 12(5):219-229.

[8]   Montenegro G, Kushalnagar N, Hui J, Culler D. (2007) Transmission of IPv6 packets over IEEE 802.15.4 networks: RFC 4944.

[9]   Parsaei MR, Mohammadi R, Javidan R. (2017) A new adaptive Traffic Engineering method for Telesurgery using ACO algorithm over Software Defined Networks. European Research in Telemedicine/La Recherche Européenne en Télémédecine, 6(3-4):172-179. Doi: 10.1016/j.eurtel.2017.10.003.

[10]   Parsaei MR, Javidan R, Fatemifar A, Einavipour S. (2017) Providing Multimedia QoS Methods over Software Defined Networks: A Comprehensive Review. International Journal of Computer Applications, 168(9):1-4. DOI: 10.5120/ijca2017914549.

[11]   Rowshanrad S, Parsaei MR, Keshtgari M. (2016) IMPLEMENTING NDN USING SDN: A REVIEW ON METHODS AND APPLICATIONS. IIUM Engineering Journal, 17(2):11-20.

[12]   Parsaei MR, Khalilian SH, Javidan R. (2016) A Comparative Study on Fault Tolerance Methods in IP Networks versus Software Defined Networks. International Academic Journal of Science and Engineering, 3(4):146-154.

[13]   Taheri R, Parsaei MR, Javidan R. (2016) A New Method for Optimizing Energy Consumption in Wireless Sensor Networks Using Enhanced LEACH Protocol. Journal of Engineering and Applied Sciences, 100(3):576-581.

[14]   Shelby Z, Bormann C. (2011) 6LoWPAN: The wireless embedded Internet. The wireless embedded Internet [BOOK], 43.

[15]   Ko J, Terzis A, Dawson-Haggerty S, Culler DE, Hui JW, Levis P. (2011) Connecting low-power and lossy networks to the internet. IEEE Communications Magazine, 49(4):96-101.

[16]   Brandt A, Porcu G. (2010) Home automation routing requirements in low power and lossy networks. RFC 5826.

[17]   Martocci J, Vermeylen W, Riou N, Mil PD. (2010) Building automation routing requirements in low power and lossy networks. IETF Request for Comments RFC 5867.

[18]   Pister K, Thubert P, Dwars S, Phinney T. (2009) Industrial routing requirements in low-power and lossy networks. RFC 5673.

[19]   Dohler M, Watteyne T., Winter T, Barthel D. (2009) Routing requirements for urban low-power and lossy networks. RFC 5548.

[20]   Moy J. (1997) OSPF version 2.

[21]   Oran D. (1990) OSI IS-IS intra-domain routing protocol.

[22]   Clausen T, Jacquet P. (2003) Optimized link state routing protocol (OLSR). RFC 3626.

[23]   Tripathi J, De Oliveira JC. (2013) Quantifying load imbalance: A practical implementation for data collection in low power lossy networks. IEEE 47th Annual Conference on Information Sciences and Systems (CISS):1-6.

[24]   Nabaei A, Hamian M, Parsaei MR, Safdari R, Samad-Soltani T, Zarrabi H, Ghassemi A. (2016) Topologies and performance of intelligent algorithms: a comprehensive review. Artificial Intelligence Review: 1-25. Doi: 10.1007/s10462-016-9517-3.

[25]   Parsaei MR, Javidan R, Sobouti MJ. (2016) Optimization of Fuzzy Rules for Online Fraud Detection with the Use of Developed Genetic Algorithm and Fuzzy Operators. Asian Journal of Information Technology, 15(11):1856-1864.

[26] Gao W, Sarlak V, Parsaei MR, Ferdosi M. (2017) Combination of fuzzy based on a meta-heuristic algorithm to predict electricity price in an electricity markets. Chemical Engineering Research and Design: 1-13. Doi: 10.1016/j.cherd.2017.09.021.

[27] Kulkarni P, Gormus S, Fan Z. (2012) Tree Balancing in Smart Grid Advanced Metering Infrastructure Mesh Networks. IEEE International Conference on Green Computing and Communications (GreenCom):109-115.

[28] Liu X, Guo J, Bhatti G, Orlik P, Parsons K. (2013) Load balanced routing for low power and lossy networks. IEEE Wireless Communications and Networking Conference (WCNC):2238-2243.

[29] Dunkels A, Gronvall B, Voigt T. (2004) Contiki-a lightweight and flexible operating system for tiny networked sensors. IEEE 29th International Conference on Local Computer Networks: 455-462.

[30] The contiki operating system. Available: http://www.contiki-os.org/index.html.

[31] Parsaei MR, Rostami SM, Javidan R. (2016) A Hybrid Data Mining Approach for Intrusion Detection on Imbalanced NSL-KDD Dataset. International Journal of Advanced Computer Science and Applications, 7(6):20-25.

[32] Parsaei MR, Javidan R, Kargar NS, Nik HS. (2017) On the global stability of an epidemic model of computer viruses. Theory in Biosciences, 136(3-4):169-178. Doi: 10.1007/s12064-017-0253-2.