ISSN 2229-5518

# Comparative study of mobility support techniques for IPv6 based RPL

Sneha K, Dr. B.G.Prasad

**Abstract**— Mobility support for Low Power and Lossy Networks called as LLNs have always been an challenging research topic. The routing protocol for LLNs known as RPL (Routing protocol for Low Power and Lossy Networks) do not support the mobility of the nodes since it has been designed mainly for static network. In this paper we present various routing techniques which supports the mobility in RPL for LLNs and highlights the performance issues of each routing technique. The aim of the paper is to do a comparative study of different techniques in terms of packet delivery ratio, delay, overhead and energy consumption over standard RPL.

**Index Terms**— LLNs, Routing, RPL, mobility.

## 1 INTRODUCTION

IoT known as Internet of Things(IoT) is becoming an increasingly growing topic which is emerging as a medium to connect people across the world for trading, gaming, emailing, and conferencing and so on. The IPv6 based RPL routing protocol developed by IETF (Internet Engineering Task Force) for LLNs is mainly designed for static networks without the support for mobility[1].

The Internet Engineering Task Force which is the expansion of IETF chartered two types of working groups IPv6 over Low Power Wireless Personal Area Networks cknow as 6LoWPAN and Routing Over Low Power and Lossy Networks known as RoLL to enable the interoperability of IPv6 based wireless sensor networks with the internet[8].

In this paper we do a study of mobility support techniques for RPL. LLNs is a class of network which comprises of constrained devices having low processing power, limited bandwidth that makes the network difficult to design and maintain. Data processing is usually required on the node itself. LLNs contains dynamic topologies where link failure, node failure and node mobility are some of the issues. LLNs are sometimes deployed in harsh environments such as

industrial and potentially deployed over link layers with restricted frame sizes. Traffic patterns in these networks could be multipoint to point (MtoP), point to point (PtoP) and point to multipoint (PtoM) flows. The default RPL specification does not consider mobility in its design goals hence when there is topology changes and some nodes are mobile, number of issues appear with respect to delay, energy consumption, packet delivery ratio and so on[2]. In this paper we mention some of the mobility support techniques which has been implemented over RPL and do a comparative study of different techniques with respect to standard RPL.

The remaining contents of paper is given as follows. In section II we present some background information related to our study that includes LLNs and RPL overview. In section III we discuss about three mobility support routing techniques Co-RPL,ME-RPL and mRPL. In section IV we discuss the Cooja network simulator which is used by three techniques for evaluation. Section V discusses the performance of Co-RPL, ME-RPL and mRPL. Section VI presents a Performance evaluation table of three techniques considering various performance parameters. we conclude the paper finally in section VII.

## 2. BACKGROUND

### 2.1 Low Power and Lossy Networks (LLNs)

IJSER

Low Power and Lossy Networks(LLNs) comprises of nodes having constraints such as low memory, processing capacity and energy. These nodes are connected by lossy links which supports lower data rates, low bandwidth and low packet delivery ratio. RPL is proposed standard routing protocol for LLNs[2]. Low-power and Lossy Networks (LLNs) which is a generic term is chosen to designate networks made of highly constrained nodes which are interconnected by low speed and unstable links[9].

**2.2 RPL Overview**

RPL is designed in such a way that it operates with less memory capacity devices and in less data traffic. Every node contains information(route) about how to reach root. RPL builds a single tree in the network and it starts by building DODAG(Destination Oriented DAG). Construction of DAG requires RPL control messages. RPL is the main component in IoT( Internet of Things) and considered as the the first candidate routing protocol for LLNs.[7]. Each RPL router in a converged Low power and Lossy Network(LLN) identifies itself with a set of stable parents each of which is a potential next-hop on a path towards the root of a DODAG, as well as a considered to be a preferred parent[10. Router, which is a part of a DODAG will emit certain messages which are called as DODAG Information Object (DIO), considering link-local multicast, indicating its respective DODAG[12]. Different control messages in RPL are

i)DIO (DODAG Information Object): This message is used to

build the DODAG since it contains general information required to build the DODAG. Some of them are DODAG_ID, DODAG_Version_Number, RPL_Instance_ID, node rank, the objective function[1-6550].

ii) DIS(DODAG Information Solicitation): A node if wants to join a DODAG it uses this message.

iii)DAO (Destination Advertisement Object):It is used to propogate destination information upward along the DODAG to the root .Each node will send the details of its parent to the root with the help of this information, the root will construct the path from source to destination.

iv)DAO-ACK: This is the message that is sent as acknowledgement for the DAO message.

**RPL Parameters**

RPL has parameters such as RPL Instance ID, DODAGID, DODAG Version ID and Rank. Each parameters has its own scope.

1.RPL Instance ID: A network is uniquely identifies by this parameter. Several DODAGs can be a part of it.. RPL instances may be local or global. RPLInstanceID found in each of the DODAG is the same but with different DODAGID.

2.DODAG-ID: Each DODAG is uniquely identified by this parameter. DODAGs within the RPL instance have same RPL instance. It is identified by tuple which consists of RPLInstance ID and DODAGID.

3.DODAG Version ID: Whenever there is a change in network, DODAG updates itself and gives a new version number to it. Version number increments sequentially and this is done by root. It is identified by tuple which consists of RPLInstanceID, DODAGID and DODAG Version Number.

IJSER

ISSN 2229-5518

4. Rank: It is the relative position of a node with respect to root node. Rank is computed using objective function. Typically a root node is at RANK 0 always. Rank increments when going away from root and decrements when going towards the root[1].

## 3 MOBILITY SUPPORT TECHNIQUES

The paper [2] Mobility Enhanced routing protocol for Low power and Lossy networks(ME-RPL) identifies three issues related to mobility

1. Mobile nodes are not been identified hence lacking in optimization of routing in presence of mobile nodes.

2. The trickle algorithm which is used in inherent design for static networks to reduce control traffic performs slower than desired in case of topology changes.

3. When a node detects changes, RPL and trickle will reset the timers locally to increase response time until the topology is stable again[2]. In ME-RPL the changes to the RPL-protocol is summarized as follows

1. Explicitly advertising mobility: Here mobile nodes are identified and furthermore advertise their mobility status in control messages[2].

2. Preferred parent selection improvement: Modification is done on the parent selection algorithm so that a node will prefer a fixed node as parent[2].

3. Detecting the topology changes , querying the neighborhood faster is made possible by adaptation of solicitation messages by the mobile nodes.

4. Mobile node arrivals because of its neighborhood may result in a node rank change.

5. A fixed node is made possible to see its parent set for any changes and also some of the mobile nodes which comes into its vicinity without any impact on its preferred parent(PP).

6. The Preferred Parent (PP) change indicates, however that either a node failure or mobility occurred in the network[2]. The dynamic DIS management algorithm is given as

Step1: Set current_DAG_Reference to dag //DAG refers to Directed Acyclic Graph

Step2: Set next_dis+1 to next_dis // dis refers to DODAG information solicitation

Step3: If PP_changed is equal to 1 then set nb_parent_changed+1 to nb_parent_changed. //Increase the number of PP changes where nb refers to number and PP is prefered parent[2].

Step4: Otherwise set nb_same_parent+1 to nb_same_parent

Step5: If PP_changed is equal to 1 then check if period_dis is greater or equal to 2*I_DIS_min and if nb_parent_changed greater or equal to N_Down_DIS then // N is number

Step6:Set DIS_period/2 to DIS_period and set zero to nb_parent_changed

Step7: If period_dis less than or equal to RPL_DIS_INTERVAL/2 and nb_same_parent greater than or equal to N_Up_DIS then go to step8

Step8: Set DIS_period*2 to DIS_period and zero to nb_same_parent.

IJSER

In the paper[3] RPL routing for LLNs using Corona mechanism (Co-RPL) an enhancement of RPL using the corona mechanism that support mobility of RPL routers is discussed[3].

The objective of Co-RPL is to maintain connectivity of nodes in their DODAG while providing quality of service guarantees at network level[3]. Corona mechanism is used in Co-RPL which reduces the chances of node failures and improves the localization of nodes in motion. It targets to support mobility in RPL for mobile MWSNs(Mobile wireless sensor networks) and fill a gap in the standard specification . Also this mechanism is able to provide QoS guarantees while preserving backward compatibility with the standard specification[3].

The corona architecture used in this approach relies on the simple concept of dividing the network area into coronas. Corona is a circular region with a certain radius centered at the DAG root[3].

The three mechanism incorporated by Co-RPL are

1. The Corona Mechanism used for the computation of corona IDs

2. Algorithms for DAG root and mobile node operations.

3. Path recovery mechanisms in case of node failure[3].

**Algorithm 1 : For Corona Mechanism:**

Step 1: Set zero to C_ID(DAGroots) //C_ID refers to corona id used for detecting node mobility and triggering neighbor discovery[3].

Step 2: DAG root broadcasts DIOs

Step 3: For all mobile node receiving DIO from neighbors go to step4

Step 4: Set min(C_ID(neighbors))+1 to C_ID(mobilenode)

Step 5: Mobile node broadcasts DIO with new C_ID

**Algorithm 2 : For operation of a mobile router**

Step 1: Broadcast DIS until receive DIOs

Step 2: For all replies if C_ID(mobile node) greater than or equal to C_ID(neighbors) then set neighbor with best quality to best parent[3].

Step 3: Else neighbor node will discard DIO and if C_ID(neighbor) changes then broadcast DIOs

Step 4: Else wait until the timer expiration[3].

**Algorithm 3: Path Recovery Mechanism**

Step1: If data forwarding to parent node by mobile node not possible then go to step2

Step2 : send the data to the node having high corona level

Step3: Inform child nodes to stop sending packets through DIS messages

Step4: Continue forwarding the data messages through neighbors by RPL router until new candidate is found[3].

IJSER

In the paper [4] mRPL:boosting mobility in IoT shows that handling changes in network topology using native algorithms, the trickle and neighbor discovery algorithm which behave in a reactive fashion are not prepared for the dynamics inherent to nodes mobility and hence mRPL uses a proactive handoff mechanism which is also known as dubbed smart-HOP within RPL.

This method shows that when the nodes are mobile packet loss reduces to half, handoff delay is also reduced compared to standard RPL. The smart-HOP algorithm has two main phases i) Data transmission Phase ii) Discovery Phase.

mRPL also describes additional timers which could improve the efficiency and reliability of the hand-off process. Algorithm 4 and Algorithm 5 briefs about the use of the timers which are used during the data transmission and discovery phase.

**Algorithm 4: Data Transmission Phase**

Step 1: If DIO packet received then Reset $T_C$ // $T_C$ refers to

connectivity timer.

Step 2: If ARSSI less than $T_l$ then go to the Discovery Phase// $T_l$ refers to threshold level and ARSSI is the average received signal strength indicator.

Step 3: Else continue the data transmission phase

Step 4: Otherwise if $T_{md}$ expires then Reset $T_{md}$ , unicast burst of DIS then go to Step 1 // $T_{md}$ refers to mobility detection timer.

Step 5: Else if Tc expires then go to the Discovery Phase

Step 6: End

**Algorithm 5: Discovery Phase**

**Step 1**: If unicast DIS message received then store RSSI readings and store counter value C of the latest DIS packet and reset Tr with (ws-C) x $T_{DIS}$

// ws refers to window size, C refers to the counter of DIS packets within each window size and $T_{DIS}$ refers to DIS interval

**Step 2:** If Tr expires then calculate average RSSI and send unicast DIO message with average RSSI // Tr refers to

**Step 3**: otherwise continue discovery phase and end

Step 4: Else continue data transmission phase and end

## 4 COOJA NETWORK SIMULATOR

The paper[2][3][4] uses the Contiki's Cooja network simualtor for performance evaluation. he routing protocol uses Contiki's modular IPv6 routing interface[11]. Cooja network simulator is a powerful tool for contiki

ISSN 2229-5518

development .It is an extensible java based simulator capable of emulating Tmote Sky nodes[5]. It allows small and large networks of motes to be simulated.

Cooja allows developers to test their code and systems long before running it on the target hardware. Cooja is also called as an emulator.

Extendibility using interfaces and plugins are the main design goal of the Cooja Simulator . An Interface represents property of a device basically position of a node, button or radio transmitter. Interaction with a simulation is done using a plugin[6].

## 5 PERFORMANCE EVALUATION

The objective of the study is to compare ME-RPL, Co-RPL, mRPL over standard RPL considering following metrics Packet Loss Ratio, Energy Consumption, Average hop count, End -to -End delay, route stability,

**i) ME-RPL Performance Evaluation:**

Performance of ME-RPL is compared with standard RPL with respect to packet loss ratio and route stability.

**Packet loss ratio**: Considering 6 to 9 nodes packet loss is always lower with ME-RPL than with standard RPL, no matter is the number of total nodes and mobile nodes present in the network[2]

**Route Stability:**

Route stability is computed as $\mathbf{S(\%)= (X_t - X_p)M_p / NM_s * 100}$

where $\mathbf{N}$ is the number of nodes within the networks, $\mathbf{M_p}$ the packet inter-arrival period, $\mathbf{M_s}$ the whole simulation time , $\mathbf{X_t}$ and $X_p$ gives the total number of sent packets and lost packets in the network.

The percentage of stable routes is always higher with ME-RPL than with RPL. ME-RPL increases the route stability [2].

**ii) Co-RPL Performance Evaluation:**

Performance of Co-RPL is compared with standard RPL in terms of Packet loss ratio, energy consumption, average hop count and end-to-end delay. These metrics are measured on three different parameters which are Packet transmission, number of DAG roots and node speed[3].

**Packet Transmission:**

**Packet loss ratio** in Co-RPL show that the packet loss ratio decreases slightly with the increase in network size. Co-RPL experiences lower packet loss than RPL(between 20% and 45%)[3]. To recover from node failures Co-RPL implements a path recovery mechanism which is flexible than RPL. Considering the transmission rates of Co-RPL it is found to be stable with less packet loss[3].

**Energy Consumption:** Co-RPL consumes nearly 50% less energy when compared to RPL.

**Average Delay:** The end- to-end delay is defined as the total time taken by a packet from node to sink. Co-RPL shows an average delay approximately 2.5 seconds less than RPL. This is because of the forwarding mechanism in Co-RPL[3].

**Impact of the number of DAG roots:**

**Packet Loss Ratio:**

In RPL Packet loss ratio decreases by 10% because of nodes getting closer to the sink and thus reducing the number of hops between nodes and their DODAG roots [3]. In Co-RPL packet loss significantly decreases and reaches 0.1 for 3 DAG roots, which implies that more number of DAG roots enhances the performance of MWSN [3].

**Energy Consumption:**

More DAGs results in reduced energy consumption (for RPL as well as Co-RPL).

**Average Delay:**

For network of large sizes the average end-to-end delay decreases when the number of DAG roots increases. The presence of multiple DAG roots reduces the number of hops between the RPL router and the mobile sink and thus minimizing the number of hops (for both RPL and Co-RPL)[3].

Lower delay of up to 2seconds is experienced in Co-RPL.

**Impact of Node Speed:**

**Packet Loss Ratio:**

RPL suffers from packet losses greater than 0.39% when node speed is 4m/sec. At higher speed the packet loss ratio is as small as 0.2% as maintained by Co-RPL[3] .

**Energy Consumption:**

Mobile nodes consume almost the same amount of energy when the speed varies from 1m/s and 3m/sec. There is only a small variation in energy consumption at all speeds for any network size.

**Average Delay:**

For RPL, the network delay increases with the speed of mobile nodes. At 4m/sec the delay is equal to 14.2sec for 100 nodes[3].

Using Co-RPL the average delay is reduced to 10sec for 100 nodes due to the corona mechanism that reduces hop count between each mobile node and the DAG root[3].

**iii) mRPL Performance Evaluation:**

In mRPL the algorithms are evaluated using three network topologies I) with two Access Points(APs) ii) with four Access Points(APs) iii) with eight Access Points(APs)[4] .

Based on the number of Access Points(APs) the accuracy of hand-off process is determined. When two Access points are present accuracy of hand-off process is higher while compared with four and denser access points. MRPL is able to perform an handoff within tens of milliseconds which is considered to be faster than RPL scenarios[4].

**Packet delivery ratio** is considered to be 100% in case of mRPL.

IJSER

Most of the packets are transmitted to the access point using fast hand-off process. In RPL, the Mobile Node(MN) should wait for control messages from the nearest AP[4]. A longer delay usually causes more packet losses because the mobile nodes are not connected to any AP. In mRPL, the mobile node is able to send data to the parent during the Discovery Phase until finding a new preferred parent[4].

The control message overhead of RPL when compared with the overhead of mRPL is found to be minimum. The periodicity of control message exchanges converges to maximum value in RPL during initialization phase after creating DODAGs.

For normal human walk spped mRPL is efficient. mRPL has less **overhead** in low traffic networks. Data transmission need o maintain network reliability in Low traffic scenarios. **Hand-off delay** is low regardless of network traffic and mobile node speed[4]. mRPL is very responsive to network dynamics. In RPL DAO transmission are used to identify the parent and based on the number of DAOs new connections are created. Switching between APs is calculated as 1.4% less hand-offs, while RPL reduces new connections approximately up to 63%[4].

## 6 PERFORMANCE TABLE OF ROUTING TECHNIQUES WITH RESPECT TO STANDARD RPL

| | | **Performance Parameters(W.R.T standard RPL)** | | | | |
|---|---|---|---|---|---|---|
| **Sl. No** | **Routing Techniques** | **Packet Delivery Ratio** | **End-to-End Delay** | **Overhead** | **Responsiveness** | **Energy Consumption** |
| 1 | ME-RPL | High | Low | Low | High | Low |
| 2 | Co-RPL | High | Low | Low | High | Low |
| 3 | mRPL | High | Low | Low | High | Low |

## 7. CONCLUSION

In this paper we presented three routing techniques used to support mobility over RPL and the comparative study on performance evaluation of ME-RPL , Co-RPL and mRPL out performs the standard RPL . Co-RPL further focuses on effect of the sink mobility on network performance as a future work. ME-RPL as future work focuses on a mobility detection algorithm for RPL and RPL rank calculation extensions. In mRPL 100% packet delivery rate is achieved upon mobility of the nodes. Future work of mRPL focuses on achieving better performance in terms of speed and hand-off delay.

## REFERENCES

[1] T. Winter and P. Thubert, “RPL: IPv6 Routing Protocol for Low Power and Lossy Networks,” IETF Internet-Draft draft-dt-roll-rpl.txt., vol. 3, 2010.

[2] Mobility Enhanced RPL for Wireless Sensor Networks by Ines El Korbi, Mohamed Ben Brahim; Cedric Adjih, Leila Azouz Saidane published in Network Of Future(NOF), Third International Conference on November 2012.

[3] Olfa Gaddour, Anis Koubäa, Raghuraman Rangarajan, Omar Cheikhrouhou “Co-RPL: RPL routing for mobile low power wireless sensor networks using Corona mechanism” Proceedings of the 9th IEEE International Symposium on Industrial Embedded Systems (SIES 2014).

[4] Hossein Fotouhi, Daniel Moreira, Mario Alves “mRPL: Boosting mobility in the Internet of Things[4] Hossein Fotouhi, Daniel Moreira, Mario Alves “mRPL: Boosting mobility in the Internet of ” published in Elsevier journal AdHoc Networks during March 2015.

[5] Voigt T, Contiki Cooja Crash Course. Swedish Institute Computer-Science.HTTPS://WWW.SICS.SE/~THIEMO/SENIOT09CCCC -SLIDES.PDF

[6] “Cooja Simulator.” [Online]. Available: http://www.sics.se/ fros/cooja.php.

[7] Olfa Gaddour, Anis Koubaa, Shafique Chaudry, Miled Tezeghdanti, “Simulation and Performance Evaluation of DAG Construction with RPL” in third International conference on Communications and Networking on April 2012.

[8] Nguyen Thanh Long; De Caro, Niccolo; Colitti, Walter; Touhafi, Abdellah; Steenhaut, Kris " Comparative Performance study of RPL in Wireless Sensor Networks”, published in Communications and Vehicular Technology in the Benelux (SCVT*), 2012 IEEE 19th Symposium on,* On page(s): 1 - 6

[9]J.Vasseur and A. Dunkels, Interconnecting Smart Objects with IP: The Next Internet. Morgan Kaufmann, 1edition, 2010

[10]U. Herberg and T. Clausen, “A comparative performance study of the routing protocols LOAD and RPL with bi-directional traffic in low-power and lossy networks,” in PE-WASUN. ACM, 2011, pp. 73–80.

[11]N. Tsiftes, J. Eriksson, and A. Dunkels, “ Low -Power Wireless IPv6 Routing with ContikiRPL”, published in the proceedings of 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, April 2010.

[12]Thomas Clausen, Ulrich Herberg, Matthais Philipp, “ A Critical Evaluation of the IPv6 Routing Protocol for Low power and Lossy Networks(RPL)”in 2011 IEEE International Conference on Wireless and Mobile Computing , Networking and Communications (WiMob).

IJSER