A ROUTE STABLE ENERGY AND MOBILITY AWARE ROUTING PROTOCOL

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

OF

MASTER OF TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

Submitted By

NIMISHA AWASTHI

2K18/CSE/10

Under the Supervision of

DR. RAJESH KUMAR YADAV

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahabad Daulatpur, Main Bawana Road, Delhi-110042

October 2019

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

DECLARATION

I Nimisha Awasthi (2K18/CSE/10) student of M.Tech Computer Science & Engineering, hereby declare that the project Dissertation titled "A Route Stable Energy and Mobility Aware Routing protocol for IoT", which is submitted by me to the Department of Computer Science & Engineering, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or any other similar title or recognition.

Place: Delhi
Date : 25 October 2020

NIMISHA AWASTHI

COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Project Dissertation titled "A Route Stable Energy and Mobility Aware Routing Protocol for IoT" which is submitted by Nimisha Awasthi (2K18/CSE/10) in partial fulfillment of the requirement for the award of the degree of Master of Technology (Computer Science & Engineering), is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this university or elsewhere.

Place: Delhi | **DR. RAJESH KUMAR YADAV**
Date: 25 October 2020 | **SUPERVISOR**

ACKNOWLEDGEMENT

Firstly I would like to express my deep sense of respect and gratitude to my project supervisor Dr. Rajesh Kumar Yadav for providing the opportunity of carrying out this project and being the guiding force behind this work. I am deeply indebted to him for the support, advice and encouragement he provided without which the project could not have been a success.

Secondly, I am grateful to Dr. Rajni Jindal, HOD, Computer Science & Engineering Department, DTU for her immense support. I would also like to acknowledge Delhi Technological University library and staff for providing the right academic resources and environment for this work to be carried out.

Last but not the least I would like to express sincere gratitude to my parents and friends for constantly encouraging me during the completion of work.

NIMISHA AWASTHI

ABSTRACT

The Internet Of Things(IoT) is emerging as a necessary tool for the modernization of world. With the advancement in IoT comes the new challenges in many areas. One area among them is routing. Though routing in IoT is challenging but the more challenging task is routing in low power lossy networks (RPL). There are some critical applications for example healthcare, which requires the mobility issue to be resolved. Mobility in RPL is a new research area for many scholars and number of protocols are developed for the same. Some work integrate mobility in a way that avoid any disconnection of moving node before finding new connection while other just allow fast recovery after disconnection. This work is broadly divided into two parts. The first part is analysis of various mobility associated RPL protocols. The main focus is on the energy consumption, handover delay, signaling cost and route stability of various algorithms and categorize the studied protocols on the basis of their property. The next part enhances a protocol called EMA-RPL protocol, which originally produces unstable routes in which topology change is frequent. The proposed algorithm selects new parent node on the basis of mobile node's speed and direction of motion. The proposed algorithm is compared to EMA-RPL on the parameters like handover delay, route stability and energy consumption.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

The internet has become an integral part of our life. With the development of smart appliances comes in the technology IoT(Internet of Things)[1] which is expanding in every area possible. With development and enhancement in IoT many global organizations has worked to provide guidelines, rules and specification to create advanced IoT applications that can improve our life. A vast range of applications are now part of IoT. Most of the devices in IoT are low battery, mobile objects that have limited resources. To achieve desired result, a well integration of mobility and resource utilization is required. Wireless sensor networks [2] is one among those areas. Wireless sensor network is considered as a set of resource constrained devices which are dispersed in a confined physical area and collects information about surroundings. All the node in WSN send the information to a central repository where analysis is performed. Wireless sensor networks has played a vital role in many of the applications like healthcare, traffic control, military etc. Since all these applications require motion of resource constrained devices, it give researchers a new area to work upon. The network type of WSN is a resource constrained called Low power and Lossy Networks(LLN)[3] and routing and resource utilization in such kind of network has always been challenging. In these types of networks all the equipment like router and interconnection are resource constrained. These types of networks are characterized by low throughput, low processing speed, high data loss and limited memory. The end devices in these types of networks mainly collect data and send it to a central repositories for analysis. Since it requires sending of data from end devices to central repositories, appropriate routing methodology need to be developed to maximize the resource utilization and avoid data loss.

Mobility[4] in any network provide many advantages by introducing more innovative services. However, mobility brings in some new challenging issues[5] that need an appropriate approach to resolve. Initially the RPL protocol was developed for low power device which do not take into consideration the mobility. It uses the strategy of self-healing[6] which raise issue like data loss, increased handover delay, route instability, frequent change in network topology and increased flow of control messages. Many protocols are developed which tries to minimize these issues. Most of these mobility integrating protocol can be categorized either as reactive protocol or proactive protocol. In proactive protocols a new parent node is found before a mobile node gets disconnected from its parent while in reactive protocol a parent node is found after disconnection but the handover delay is reduced. Most of the proactive protocol uses RSSI (Reduced Signal Strength Indicator) value to check for the mobility and once the motion is confirmed, a new parent node is found. Most of the reactive protocols uses reduced trickle timer interval to handle mobility. By using a reduced trickle timer, a mobile node is able to find new parent speedily as compared to classic RPL. In order to handle this mobility problem, the RPL may react through a self-healing strategy. The result produced using this strategy do not provide the desired result. To overcome this shortcoming protocols are developed to integrate mobility with efficiency. With these developments the various important parameters like handover delay, energy consumption, route stability are improved.

1.1 RPL OVERVIEW

Routing in Low power Lossy networks (RPL) is a distance vector routing protocol for resource-constrained networks that forms a tree-based Destination Oriented Directed Acyclic Graphs(DODAG) and aims to establish a seamless network. The RPL consists of control messages; DIO, DAO, and DIS[19] which are used for the construction of DODAG tree.

- DAG Information Object(DIO): In RPL for the construction of paths from root node to leaf node DIO messages are used. DIO messages are mainly for DODAG discovery and maintenance. Each Node on expiry of a defined trickle timer transmit a DIO messages periodically.

- Destination Advertisements Object (DAO): DAO message is used to transfer traffic from root to downward in non-storing mode. In this case, if a node is having some data to be sent to some other non-root node then data is first transferred to root node and then root node transfer the data to destination node. DAO message is send by a node to its parent node.

- DAG Information Solicitation (DIS): DIS message is designed to be sent by a new node to join the DODAG.

The DODAG building process is started by root node or sink node. The root node sends DIO message to all the neighboring nodes. The DIO message contains Objective Function (OF) [7] and other details. Each neighboring node on receiving DIO message decides whether it want to join DODAG or not based on OF. A DIO message can be received from more than one node, the parent node selection is done based on node rank. The new nodes, which joins the DODAG, construct new DIO message by calculating their node rank and send it to their neighbor node. In this way the whole DODAG tree is formed. When a new node wants to join the DODAG it sends a DIS message to all the neighboring nodes. In response, the new node receives DIO message and according to the parameters, it decides which node to select as parent.

The RPL DODAG tree construction is shown in figure 1.1. As shown in the figure first the root node sends the DIO message to all the neighbor nodes and so on until the tree is constructed. In the fig DIO message to only those node is shown whose parent is not selected but in actual DIO message is sent to all the neighboring node. Since RPL is for low power networks, it must ensure that energy consumption is low. In order to achieve so RPL uses trickle timer [8] to slow down the control message.

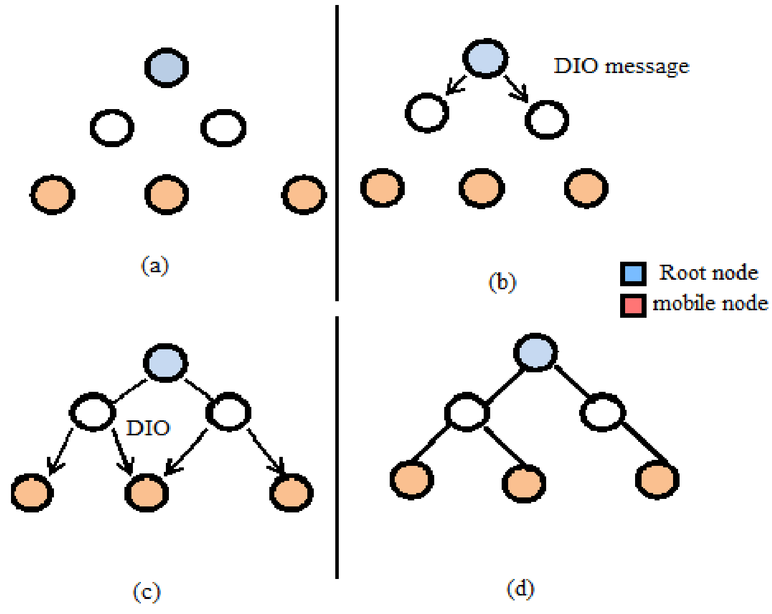

Figure 1.1 RPL DODAG construction

1.2 RSSI OVERVIEW

Received Signal Strength Indicator (RSSI) value is calculated by the parent node using the messages received from child node. It gives the strength of signal received and is measured in decibel. The range of this value varies from 0 db to -100 db. The closer the value to 0db the better the signal strength. Generally a RSSI value of -60 db is considered to be a good signal strength. The more the value is negative the weaker the signal strength and higher the data loss. Most of the mobility integrated RPL protocol uses this value to get reliable paths.

`

CHAPTER 2: LITERATURE REVIEW

RPL was originally designed for low power lossy networks and it was assumed that network topology change would be minimal, but due to mobility an inconsistency arise in topology. When this happens, the mobile node becomes detached from it's all the connected nodes (parents and children). In this scenario, RPL react with self-healing strategy by locally repairing the path by receipt of DIO message after the trickle time. In this way in order to avoid data loss, RPL tries to find an alternative path which is not optimal.

2.1 RELATED WORK

In Mobility Enhanced RPL (ME-RPL) [9], unlike traditional RPL firstly a distinction is made between static nodes and mobile nodes. Mobility is included in the control messages using flag to increase route stability and avoid unnecessary disconnection of routing paths. Mobile node can be only leaf node and never a parent node. Further to improve recovery from the disconnection due to mobility, trickle timer period is kept dynamic. It depends on the mobility frequency of the network. The higher the mobility in network the less the trickle timer period and accordingly DIO messages are triggered. Sometimes due to high mobility many DIO messages are triggered which may cause high energy consumption of the node. It is a reactive protocol as it finds new parent node after the child node has lost its connection. Many other parameters like handover delay[10] which is very important for many critical applications, route stability and others are also not considered.

In [11] the authors enhances and extends the MERPL protocol. EMA-RPL starts with the same condition as of [9] that a node which is moving will be distinguished from a static node using mobility flag but the approach used in this is very different from that of [9] as it provides a proactive protocol as before the leaf child node gets disconnected from its parent node due to motion, the parent node gets to know about this mobility using RSSI [12] value. This value is calculated from the messages received from leaf child node and if this value is not above a threshold value then parent node is alerted and before disconnection, it starts finding new parent node. The entire process is carried out by the parent node itself that is from discovering a new parent after detecting mobility to notifying about the new parent to mobile node, hence it saves energy consumption of mobile node. In terms of handover delay, energy consumption and other parameters it beats both RPL and ME-RPL but the routes generated using this protocol are highly unstable. There is low route stability as the new parent node is selected when the node is still in motion.

An Extended KP is used in [13] which gives routes that are more stable as compared to [11]. The approach is similar to EMA-RPL up to some extent as mobility detection is same as that of EMA-RPL and beside this it is also a proactive approach, differentiating between mobile nodes and static nodes. The difference is that the new parent node is found by the mobile node itself using its co-ordinates and direction of motion. The co-ordinates are enhanced using Extended Kalman Filter. The route stability as compared to EMA-RPL is high and disconnection rate is reduced. The thing of concern is that most of the calculation and updation are performed by mobile node itself which increase mobile node energy dissipation.

A very unique approach is used in [14] which uses corona to deal with mobility issue. In this, the author have not applied any constraint that a mobile node need to be a leaf node. A mobile node can also be a parent node. Some fields are added for Co-RPL in default RPL control messages to integrate mobility with minor modification. There are many DAG roots which are considered to be static and DAG building starts with them. Each DAG root forms a corona. The corona ID of root node is selected as zero. Upon receiving DIO message, the node selects the best parent using lowest corona ID, then determine its own corona ID by adding up one in the lowest corona ID of the received neighbor's message. After calculating the corona ID, it updates the DIO, and broadcast it to neighbor nodes. When due to mobility a node is unable to reach its destination then data packet is send to any node with upper corona ID and the parent node ask its children node to stop sending data. In case of receiving a DIS message before expiry of trickle timer, DIO message is sent immediately before the expiration of trickle timer. The advantage of this mechanism is the packet loss ratio reduces to a considerable number with low energy consumption and reduced end to end delay. Since it allows the parent node to be mobile, route instability increases which not only affect its own data but other child node's data also.

In order to provide mobility support to networks in which data traffic varies highly, the author in

Backpressure RPL (BRPL) [15] proposed the merging of RPL and the concept of backpressure routing. BRPL allow multiple logical DAG to be created to handle mobility and traffic. To achieve the desired result for each DAG, a buffered packet queue is maintained by each BPRPL node. Since there are many DAG there are many routes from one source to destination hence when a route is disconnected due to mobility or when there is congestion in a particular route another route can be followed to improve QoS. In order to handle mobility a parameter called theta θ is used whose value ranges between zero to one. The value of this parameter is adjusted dynamically by using QuickTheta algorithm that calculates the θ value based on parameters like congestion level and the mobility of nodes. The advantage is that it reduces the packet loss but it increases the end-to-end delay.

Another approach similar to the approaches using only fixed node as parent node is used in [16]. The author unlike traditional RPL which selects the parent node on the basis of least rank and Euclidean distance, also considers other parameter for the selection of parent node. A preferred parent node set is first constructed using node rank and Euclidean distance (priority given to node rank) , and then among the preferred parent list the best parent is selected using static or dynamic status, ETX_min, ELT_max, and RSSI_max obtained from DIO message. In order to look for mobility a D-trickle algorithm is used in which the trickle timer is dynamically calculated using neighbor nodes. It is a simple and efficient algorithm to deal with mobility. It supports mobility with no change in RPL control message and increases packet delivery ratio. The disadvantage is it may not give satisfactory end to end delay and it is reactive protocol.

In order to manage downward traffic under mobility the author in [17] assumes that the new parent selected would be any neighbor node only. Hence in the absence of ACK the parent node broadcast the data destination to mobile node to all the neighboring nodes. The new parent selected will transfer the data without delay. This algorithm definitely improves downward traffic management but it has a drawback that broadcasting effects the energy consumption of network and also causes unnecessary flooding of messages. The assumption that the new parent node would be a neighbor node is also not always true.

A route stable strategy is presented in [18] called KP-RPL. It is also a proactive protocol and like any other of this type it also differentiate between mobile nodes and static nodes. The mobile node is never allowed to be a parent node. A list of blacklist nodes is created by all the mobile nodes, which includes static nodes having low RSSI value and can possibly produce unreliable routes. The route produced in this algorithm are quite stable. However the assumption of low mobility make this protocol restricted to some areas only. The energy consumption of mobile node is also high as most of the calculations are performed by mobile node itself and even sometimes positions are not accurately calculated.

An integration of RPL with smart hop is presented in mRPL [19]. The author have proposed a solution to enhance mobility in RPL by actively reacting to any disconnection in the network. Various timer like connectivity timer,mobility detection timer, handoff timer and reply timer are used to handle mobility. The algorithm is broadly divide into two phases as data transmission phase and Discovery phase. In data transmission phase a mobile node based upon the value of average RSSI and various timer value decides whether to send data or stop sending data and enter the next phase. In discovery phase, new parent node is detected using ARSSI value. This value is calculated by every neighbor node from the DIS burst broadcasted by mobile node and a unicast DIO message is sent by neighbor node to mobile node. On the basis of ARSSI value received from all the neighboring node, the node with best ARSSI value greater than a threshold is selected as parent node or access point. This protocol can neither be categorized as proactive or reactive protocol as if it comes know about weakening of a link, it immediately stop sending data to avoid any loss and can be called a proactive protocol but if this disconnection is not detected at an earlier stage then it act as reactive algorithm. Sometimes the handover delay is considerable as a DIS burst is sent by mobile node and unicast DIO message are replied in a non-colliding manner which sometime may increase handover delay.

In order to improve RPL for mobile environment, the authors in [20] proposes D-RPL, a routing protocol for IoT applications which are dynamic. This algorithm dynamically adjust the trickle time period on the basis RSSI value received from the messages. Hence it deal with mobility in reactive mode. In order to maximize the responsiveness and smooth transitions and minimize handover delay it includes some routing metrics in calculation of trickle timer. This same approach with some modification is adopted in [21]. (GTM-RPL) like any other proactive approach uses received message RSSI value to detect mobility. A game is developed in which nodes act as a player and compete to send data.

An additional layer which is used for routing is introduced in [22]. All the routing messages passes through this layer while other messages containing data do not go through this layer to avoid any overhead. This algorithm uses fuzzy estimator to find the best parent. Information about the neighbors is fed into fuzzy estimator and on the basis of output route is established. The link disconnection is detected by the new layer introduced using the count of retransmissions. The mobility is handled either by reconstructing the default route or by sending a message to non-root destination about disconnection. The described algorithm is reactive protocol which allow a mobile node to be parent node and sometimes handover delay is increased considerable compared to RPL protocol.

To support mobility in healthcare author in [23] proposed a simple adaption of RPL named mod-RPL(modified RPL). This algorithm just avoid mobile node to be a part of routing path. In order to achieve so mobile nodes are not allowed to send a DIO message and hence it can be only leaf node in DODAG tree. It is a very simple protocol that do not take into account some of the important and complex issues like handover delay and route stability. As healthcare applications are time critical, they need a proactive protocol that can find a new connection before getting disconnected. This is a simple approach which regulates the transmission of control messages but it assumes mobility is not so high which is also a drawback.

In [24] ,to handle mobility the author has designed a timer algorithm. The preferred parent is selected on the basis of lowest rank which is calculated using ETX (Expected number of transmission) and RSSI value. A value called time to leave(TL) is calculated which gives the amount of time it will require for the mobile node to be out of range of the parent node. Initially DODAG tree is constructed by using default RPL method and then routes are refined by using RSSI value and avoid any mobile node to be a parent node. According to the timer set the frequency of DIO message is changed.

The proposed algorithm in [25] reduces the power consumption of mobile node by dynamically changing the DIO timer. Compared to RPL it also reduces the handover delay by reducing the DIO timer when a high mobility is detected using a quantity called mobility level(ML). Unnecessary flooding of control message is also reduced when a low change in network topology is observed by increasing the DIO timer. Though it successfully reduces power consumption effectively but is a reactive protocol. Data loss might also occur due to this.

A demo model [26] is proposed which uses four mechanism to support mobility in RPL. First is the differentiation between mobile node and fixed node. The second is to avoid selecting disconnected routing path by adaptive timing & probing. The third uses two Metrix as RSSI and hop distance for parent selection and the fourth is proactive neighbor discovery so that a new route is found when needed. This algorithm is a combination of some of the popular mobility enhanced RPL. It would have been much better if the proposed method have also included route stability and handover delay.

An algorithm similar to the algorithm in [11] is described in [27]. The entire process is almost the same with some differences. In the later one for parent selection three parameters are used as ETX, RSSI, residual energy. The new parent selection steps in this algorithm is performed by mobile node itself after the mobility is detected using DIS message and hence consumes more power of mobile node. Route stability issue exists in this algorithm also.

A content Centric based routing is proposed in [28]. The algorithm is roughly divide into two parts. The first part deals with mobility enhancement which is carried out by assuming that the new parent node selected would be neighbor of old parent node. Hence when a connection is lost the data for mobile node is broadcasted to all the neighbor node. The second part introduces content centric approach for power consumption and traffic management. A mobile nodes is distinguished from other static nodes using a flag. The proposed algorithm is reactive in nature and the assumption that new parent node is always a neighbor of old parent node give unstable routing path.

2.2 COMPARISON OF STUDIED PROTOCOL

In this section we analyze various protocols. Figure 3.1 shows the energy consumption of above discussed four protocols as KP-RPL, EMA-RPL, ME-RPL, RPL. It can be seen that KP-RPL power consumption is more as compared to EMA-RPL for mobile node.

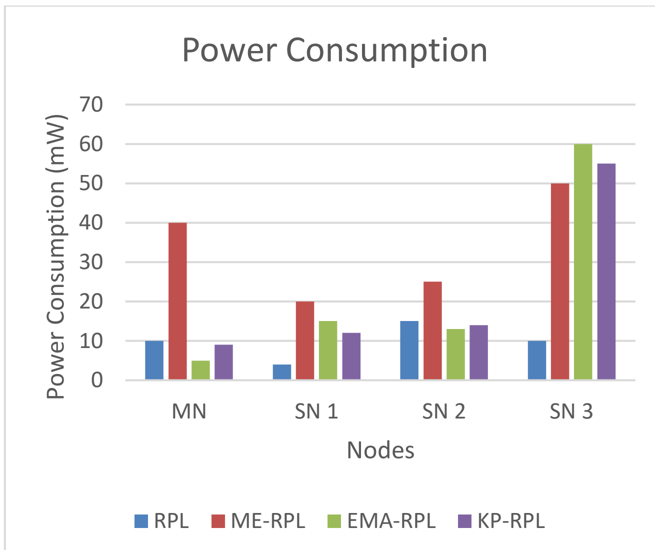

Figure 2.1 Energy consumption of some of the protocols discussed

Route stability of some of the proactive protocols discussed above is shown in figure 3.2. The less the topology change the more the route stability is in the network and the smoother the routing. As can be analyzed from the figure that KP-RPL and EKF have nearly the same route stability and are the best comparing to other algorithms. In some cases EKF outperforms KP-RPL as it filters the position of the mobile node using enhanced Kalman filter.

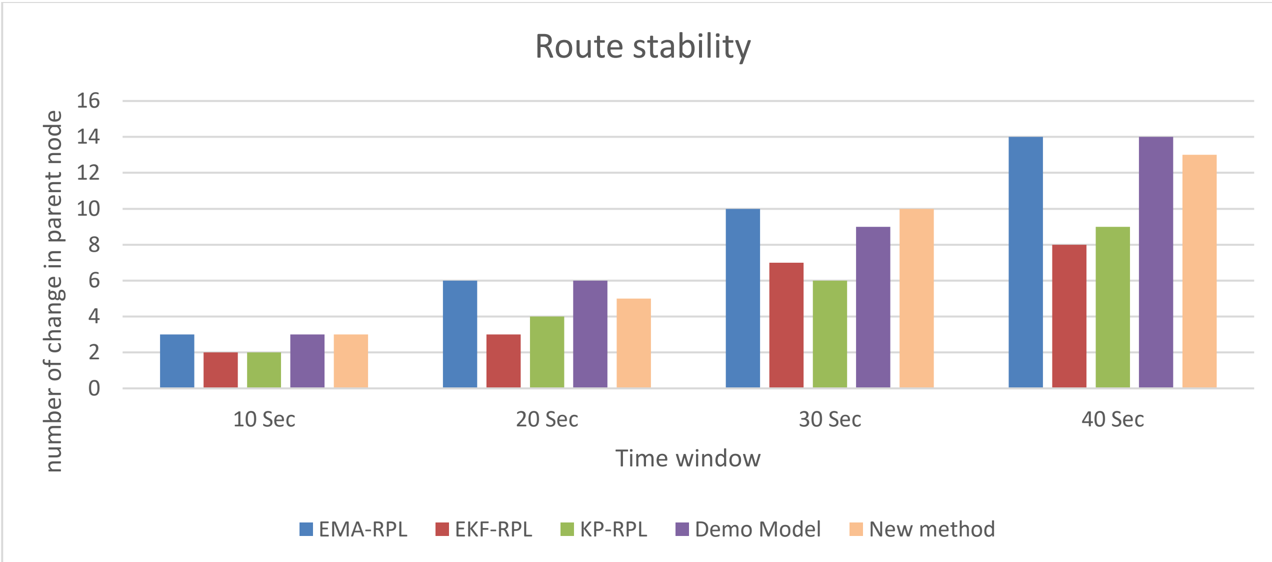

Figure 2.2 Route stability of proactive protocols

It can be inferred from the figure KP-RPL and EKF-RPL gives more route stability as compared to other routing protocol in same category.

The most important aspect of routing in LLN is energy consumption. As from above discussions some protocols have managed energy consumption of the network effectively while some still lack behind. Another important issue is route stability which some protocols have managed efficiently. Some of the proposed protocol like those described in [17] and [28] integrate one part of the mobility which manages downward traffic in DODAG. Other described protocol do not look into this aspect. Algorithm in [14] and [22] also somewhat deal with downward traffic but to some extent only. Algorithms in [13], [15] and [18] provide more route stability compared to other algorithms as they select the new parent node based on some conditions and metrics.

One of the important characteristic of any mobility integrated protocol is whether the protocol is reactive or proactive. In proactive protocols a new parent node is found before a mobile node gets disconnected from its parent while in reactive protocol a parent node is found after disconnection but the handover delay is reduced. The second important aspect of the mobility is whether a mobile node is allowed to be a non leaf or node. The higher route stability is achieved when a mobile node is not allowed to be a parent node. Table 3.1 gives some of the basic information discussed above with limitations about the studied protocols.

Table 2.1 Limitations of protocols

Protocol	Limitations	Mobile node allowed to be non leaf node	Reactive/ Proactive
ME-RPL [9]	Increased handover delay	No	Reactive
EMA-RPL[11]	Unstable routes	No	Proactive
EKF-MRPL[13]	Consume more energy of mobile node	No	Proactive
Co-RPL[14]	Allow mobile node to be parent and hence increases route instability	Yes	Reactive
BRPL[15]	Increased end to end delay	Yes	Reactive
MAEEPS-RPL[16]	Reactive and increased end to end delay	No	Reactive
KP-RPL[18]	Mobile node resources are wasted.	No	Proactive
D-RPL[20]	Reactive and unstable routes	Yes	Reactive
MoMoRo[22]	Increased handover delay	Yes	Reactive
Mod-RPL[23]	Handover delay is not considered.	No	Reactive
Adaptive Timer RPL[24]	Increased flow of control messages.	Initially yes, later No	Reactive
Improved Power Consumption[25]	Date loss.	Yes	Reactive
Demo model[26]	No new improvement.	No	Proactive
New method to improve RPL[27]	Route instability and consumes more resources of mobile node.	No	Proactive

Content Centric Routing In RPL[28]	Assumes new parent node will be a neighbor node.	-	Reactive

` ## 2.3 CATEGORIZATION OF STUDIED PROTOCOLS

Most of the work studied in this survey can be categorized based on the way mobility is handled. Broadly most of the methods can be classified as either proactive or reactive protocol. Some methods which deal with downward traffic are not categorized among these two categories because of the entirely different approach and goal. Further most of the reactive protocol rely on trickle timer to handle mobility while some can be categorized into miscellaneous or other types which do not change the trickle timer duration. Figure 3.3 shows this categorization of the studied protocols. mRPL is not categorized in these categories as its behavior changes according to received information.

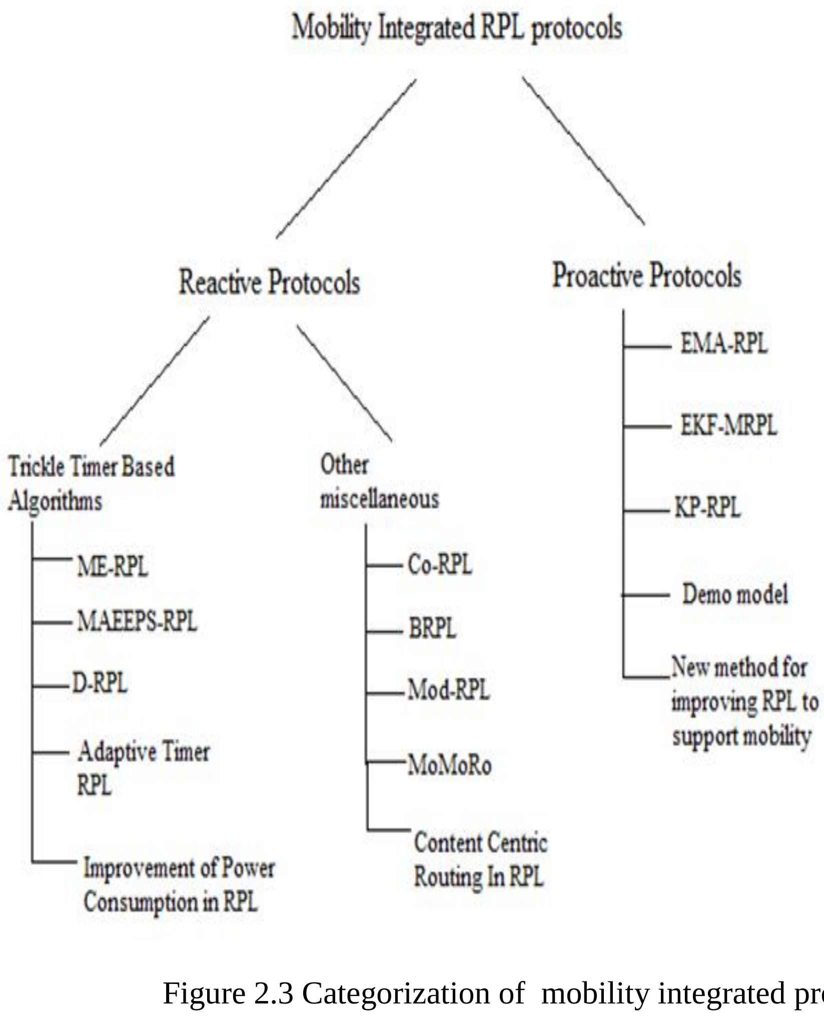

Figure 2.3 Categorization of mobility integrated protocols

CHAPTER 3: PROPOSED ALGORITHM

3.1 OVERVIEW OF EMA-RPL AND THE PROPOSED PROTOCOL

The foremost purpose of this proposed work is to reduce the use of mobile node resources and enhance the stability of EMA-RPL pathways. EMA-RPL can be classified as proactive protocol that allows disconnection of mobile node to be determined before network disconnection, which solves the problem of RPL frequent topology changes. This will reduce the number of EMA-RPL modified preferred parents. Most of the calculations are performed by fixed nodes, so the power dissipation of the mobile nodes is reduced. Depending on the mobility of the mobile node, the handover delay may increase. The procedure used to find a steady path depends on the velocity of the node in motion and the direction. The new presented protocol enhances mobility, reduces data loss and handover delays, and improves path stability.

3.1.1 Outline and changes in existing Energy and mobility aware RPL

Energy mobility aware RPL can be defined as an improved version of the RPL protocol that integrates RPL mobility with minimal loss of messages and delivery delay. With this proposed methodology in this protocol, nodes which are not in motion could contain children nodes and all the moving nodes are terminal nodes or leaf node.

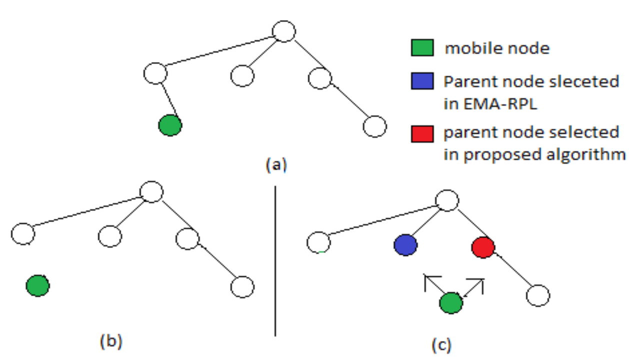

Figure 3.1 Selection of parent in both existing and presented protocol

The exact goes for the proposed work, where all mobile nodes become leaf nodes, minimizing data and path loss. Energy Mobility Aware RPL can be roughly divided into described three phases. The first stage is the detection and transmission stage. At this point, the motion of mobile node is detected by linked nodes, also known as preferred parents, with low RSSI (Received Signal Strength Index) values.

Enhanced EMA-RPL static path mobility detection follows the exact rules as that of EMA. The next stage of EMARPL is the reaction phase with prediction included where the new ancestor node is discovered. For this step of the presented methodology, there are some additional modification to the ICMP control messages. The final stage of Energy mobility aware RPL is the apprising phase where the mobile leaf node gets informed about the new assistant node. With the new protocol, notifications changes in response to changes in phase two.

3.1.2 Mobile node's speed calculation

The path stability of the presented algorithm can be attained with the help of velocity of the mobile node. Speed can be calculated with the help of changes in the co-ordinates of the moving node. If the primary coordinates of the mobile node are considered to be (x_i, y_i) and after some Δt unit of time, the new coordinates observed of the moving node are (x_m, y_m)

Scalar Component of Velocity along x-axis $(v_x) = (x_m - x_i) \,/\, \Delta t$ (3.1)

Scalar Component of Velocity along y-axis $(v_y) = (y_m - y_i) \,/\, \Delta t$ (3.2)

Speed $(v) = \sqrt{v_x^2 + v_y^2}$ (3.3)

3.1.3 Modification in ICMP message

Several changes have been made to ICMPv6 control messages, including EMA-RPL changes, to include path stability in the EMA-RPL. In addition to EMA-RPL changes such as including RSSI values and control flags, some other fields are attached to ICMPv6 (DIS, DIO) control messages such as mobile node coordinates.

3.2 THE PROPOSED ALGORITHM

After configuring the DODAG tree for the connection, the mobility is determined using RSSI value (a reduction is observed). When motion is determined, the associated parent node of the mobile leaf node (also known as the assistant node in the EMARPL protocol) begins to discover a suitable associated parent node. The presented algorithm go in the beginning of the next stage. This is to search for a mobile node an another associated parent node.

3.2.1 Identification of new parent node

The linked parent node broadcasts a DIS message containing the RSSI value with the *flag* set to 1 and the mobile node identifier (ID). The clock and coordinates of the moving node are also recorded for velocity calculations when the DIS message is broadcasted. Assume the clock recorded at time the DIS is disseminated is noted as t_0. In addition to this t_0, the timer is activated by the associated linked node which can be used to define the waiting for a response for the DIS broadcast message. DIS broadcast messages are encountered by mobile and other nearest fixed nodes. When a moving leaf node receives a DIS communication from the associated linked node, it decides whether to carry on with sending data or end to avoid data loss, based on the RSSI value of the DIS message received. The mobile node also broadcasts a DIS message in which *flag* is set to 2. This message is also detected by the static node (SN). Mobile node's broadcast messages are received by the adjacent SN and its node, so the associated node knows the present latest coordinates of the mobile leaf node. When mobile node's distributed message is encountered by the linked parent node, record the clock as t_n.

Associated linked node computes the mobile node velocity as follows.

The primary coordinates of the mobile node considered initially are (x_i, y_i) and the latest new coordinates calculated are (x_m, y_m). The time shift is t_n-t_0. The calculated velocity is represented as follows:

Velocity of the moving node = $\sqrt{\left(\frac{x_m - x_i}{t_n - t_i}\right)^2 + \left(\frac{y_m - y_i}{t_n - t_i}\right)^2}$ (3.4)

Each SN that receives a DIS communication broadcast by the moving node respond with a unicast DIO communication to the AN containing the RSSI value computed with the help of the received moving node's DIS communication. Based on the received RSSI value, the linked nodes compare them to discover the finest RSSI value. The fixed node having the highest RSSI value in scrutinized for the second step.

3.2.2 Selecting stable route

A specified node that is considered to be good parent node is notified of the highest RSSI value by the linked node that, it is selected to facade as a linked node under a few specific terms. The informing also includes velocity of the moving node movement and the period during which the velocity was calculated (t_n-t_0). Previous linked node also informs the moving leaf node of the new linked node. The newly linked parent node then sends a DIO communication with *tag* = 3 to moving node telling it that it will behave as temporary associated parent so as it can find a new stable linked path.

On encountering a DIO communication in which *flag* is equal to three, the moving leaf node respond with a one directional DIO control message to the new linked parent node, provides it's coordinates, and begins the clock. When the new linked parent node encounters a DIO from moving leaf node, it take down the mobile node coordinate and the clock $t_{a_0}$when the message was encountered. When the triggered timer run out on the moving leaf node, a message containing the coordinates is resend. Upon receiving the message, the new linked node calculate the velocity of moving leaf node is calculated in the same way as was done by the old associated parent node and then decide whether to find another parent node on the basis to new velocity computed s_n. The following can be inferred

- If the calculated new velocity s_n is larger than or equals to the old velocity s_i. It is clear then that the moving leaf node is still in a random motion and the new linked node temporarily bound and does not provide a stable path.
- The new velocity is in the range of 0.3 * s_i to s_i ,that is, between 30% of the previous velocity and the previous velocity, the new linked ephemeral node may not provide a stable path. Whether it gives a stable path depends on mobile node's speed.

- If the range of the new velocity calculated s_n is below $0.3 * s_i$ then there might be a high chance that temporary parent acting node may provide a stable path.

From the exceeding statements following rules can be inferred

- If the new speed calculated meets the initial two terms, creating a new temporarily bound node as the preferred node will not stabilize the path. Therefore, instead of creating a new associated node as the parent node, the new related node acts as a short-term master node, uses the above algorithm to find another preferred parent that is more reliable.

- If the new speed calculated of moving node meets the last term, the associated ad hoc node is selected as the preferred asset. This will notify the mobile node. When the mobile node receives the DIO message, it updates parameters such as parent, routing to the new root node, and notifies the relevant temporary node with this command.

3.2.3 Accuracy in the speed calculation at different point of times.

As with calculating the speed, the delay associated with sending a message is not taken into account, so the calculated speed may appear inaccurate. There are subsequently two possible cases in it -

Case I: When computed propagation delay is nearly equivalent.

At first when the assistant node gets to understand the moving of mobile leaf node while receiving lessen RSSI parameter value, Let the time be t_i at which the co-ordinates are noted down. Let us suppose it takes the ∂t_i time for a message to reach from the mobile leaf node to associated parent node. Then the actual clock nearly at which moving leaf node was really at the received coordinates in the communication is

Confirmed clock at which the moving node was at the given coordinates (x_i, y_i) is

$$t_i - \partial t_i \tag{3.5}$$

Similar at a stage later if ∂t_n is supposed to be the delay then real time is

Confirmed clock at which leaf moving node was at the given coordinates (x_m, y_m) is

$$t_n - \partial t_n \quad (3.6)$$

Then calculated time difference for speed calculation is

$$\Delta t = (t_n - \partial t_n) - (t_i - \partial t_i) \quad (3.7)$$

Now if the delays in the two conditions is approximately equal as assumed for the case, then time difference approximately becomes as

$$\Delta t = t_n - t_i \text{(approximately)} \quad (3.8)$$

As ∂t_n and ∂t_i are considered to be equal and gets nullified .

Hence the speed calculated is accurate and gives desired result.

Case II: The calculated propagation delay can be substantial.

If the messages from the mobile leaf node to previous associated node have different propagation delays, there will also be some differences when the newly found temporarily linked node calculates the new speed. This unequal delay exists in the both cases, so the calculated speeds at different clocks will have roughly the same miscalculation. Because of to the delusion, the actual calculated velocity will be slightly slower than the real velocity, but since this slow velocity is also in the next recorded velocity at a stage later, hence there is no big contrast and the method gives desired result with speed fallacy.

Chapter 4: Implementation

For implementation we uses an free source IoT environment system called the Contiki, which is a IPv6 / 6lowpan platform and uses the widely used freeware execution of RPL called Contiki RPL [20]. Contiki OS provides Cooja simulations that simulate different types of Contiki nodes. Contiki is used in the proposed method because it gives a built-in environment for RPL and have many types of nodes that closely mimic realistic WSNs.

The execution of the presented method was performed using the following framework defined in the cooja simulation.

Table 4.1 Parameters for simulation

Data rate	30 packets/s
Range of Transmission	50m
Threshold Fixed for RSSI	-90dBm
Execution time	800s
Sensing Area	150*150 m^2

In contiki firstly nodes are selected and the following screen appears as shown in figure 4.1.

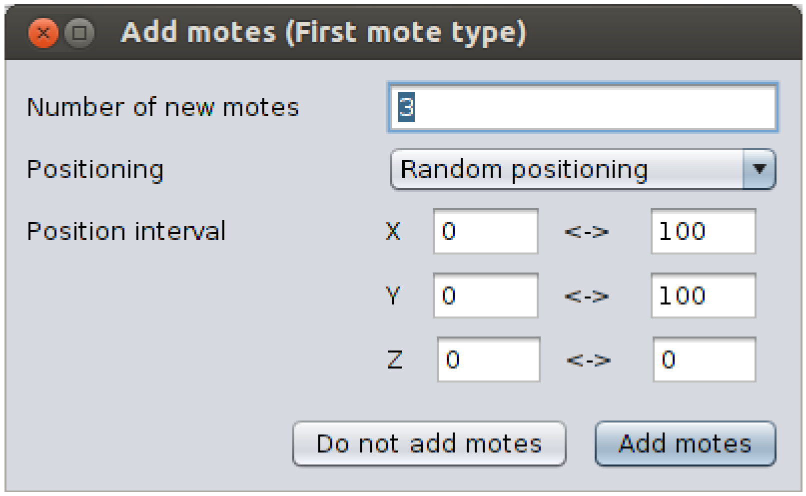

Figure 4.1 selection of motes

After selecting various parameters we start the simulation. Figure 4.2 shows simulation screens.

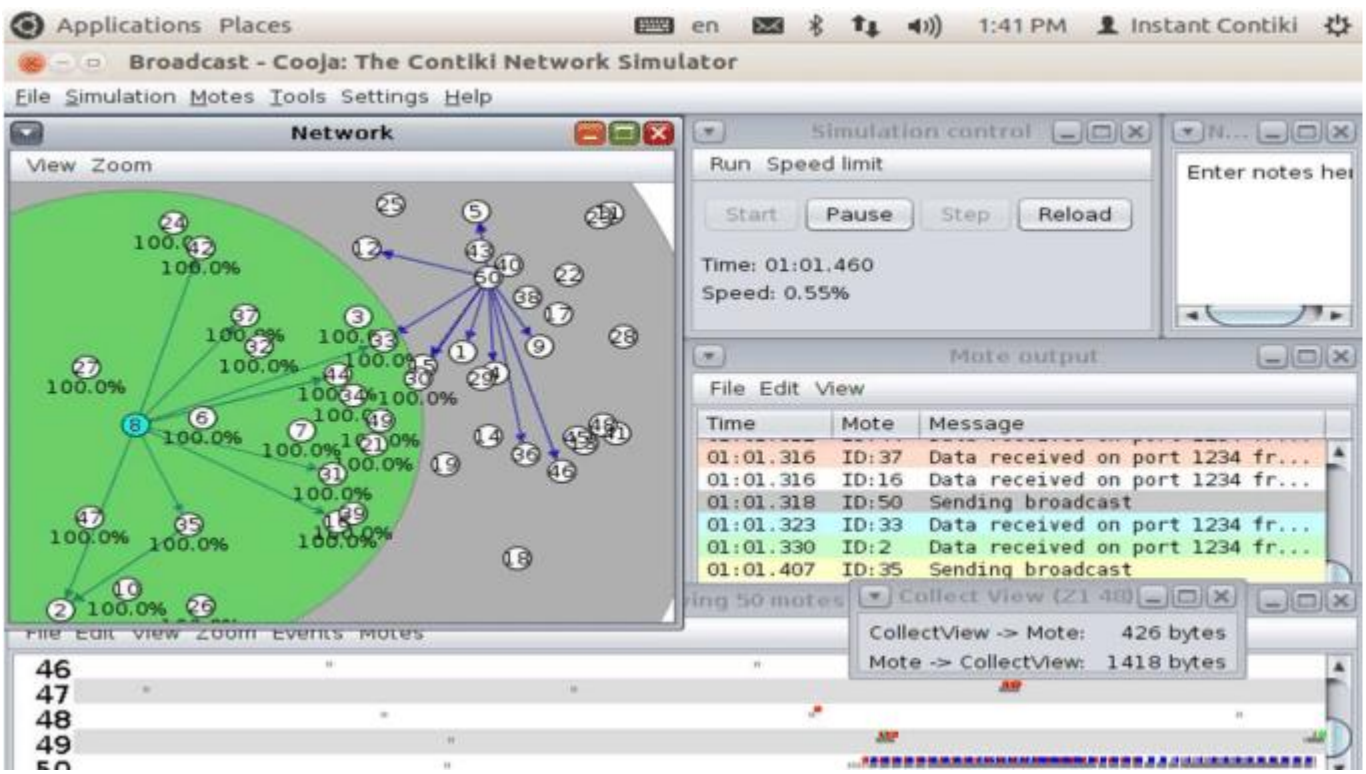

Figure 4.2 Simulation of proposed protocol

CHAPTER 5: RESULTS AND PERFORMANCE EVALUATION

5.1 RESULTS

As can be seen from the performance evaluation of the new proposed algorithm, it shows that outcome is a new more stable routing protocol which gives better result. As can be seen from the graph that route stability is increased as compared to EMA-RPL. The power consumption of any node in the new discovered algorithm is also comparable to EMA-RPL. The handover delay might seem higher but the average handover delay is nearly the same. Further the handover delay can also be reduced by changing the value of the threshold speed. The proposed protocol gives better performance for WSN.

Hence the outcome are as shown in table 5.1

Table 5.1 Result values

Parameter	**EMA-RPL**	**Route-Stable EMA-RPL**	**Outcome**
Power Consumption (mW)	20.5 (Average)	19.5(Average)	Proposed protocol outperforms EMA-RPL
Handover Delay (ms)	200 (Average)	225 (Average)	Nearly same but EMA is better
Route Stability (average parent node change)	7.25	3.25	Proposed protocol outperforms EMA-RPL

5.2 PERFROMACE EVALUATION

Our proposed new route-stable EMA-RPL is compared with the existing EMA-RPL protocol in this section. Around over five simulations were performed to get as accurate result as possible. Bothe the protocols are compared on three parameters as power consumption, the handover delay and the route stability. As EMA-RPL and proposed algorithm both have nearly same energy consumption hence power consumption comparison id done between the new algorithm and EKF algorithm.

5.2.1 Route Stability

The stability of any route can be defined with the help of rate at which the parent node changes in a given interval of time. If a mobile node changes or select S_P preferred parent in a time interval t. Then parent node change rate is defined as-

Derivative of parent node change

$$\partial P = S_p/\mathrm{t} \qquad (5.1)$$

Now the route stability is calculated using derivative of parent node change in any network. It in inversely proportional to the rate of change of parent node and is given by following equation-

Stability of a route

$$S_R = \frac{K}{\partial P} \qquad (5.2)$$

As can be inferred from above equation that the more the frequent the change in parent node the more the unstable the route is. As can be seen from the figure 6.1 that rate of change of parent node in our proposed new protocol is much less as compared to EMA-RPL and lower the value of rate pf change of parent node, higher route stability. The proposed protocol is not compared with EKF protocol because they both have nearly similar stability in routes

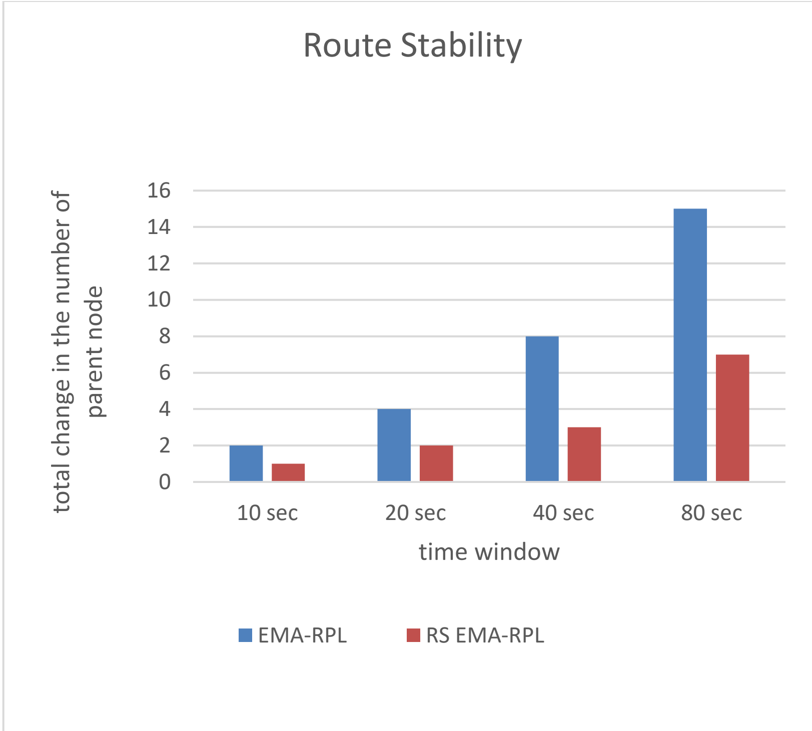

Figure 5.1 Route stability Comparison

5.2.2 Handover Delay

Handover delay is a very important parameter in any of the routing protocol. Whether it is a static or dynamic protocol, handover delay must be minimized to produce seamless routes. It can be defined as the time elapsed between the last and first message send by any node before disconnection with its parent node and after finding a new parent node. Average Handover delay can also depend upon rate with which the parent node change. As if parent node change is less frequent less time is wasted and messages are send without any waiting. Our proposed algorithm is nearly same in handover delay as compared to EMA-RPL as though for a single message it might be more than that of EMA-RPL but on average it is approximately the same as parents in proposed protocol changes less frequently.

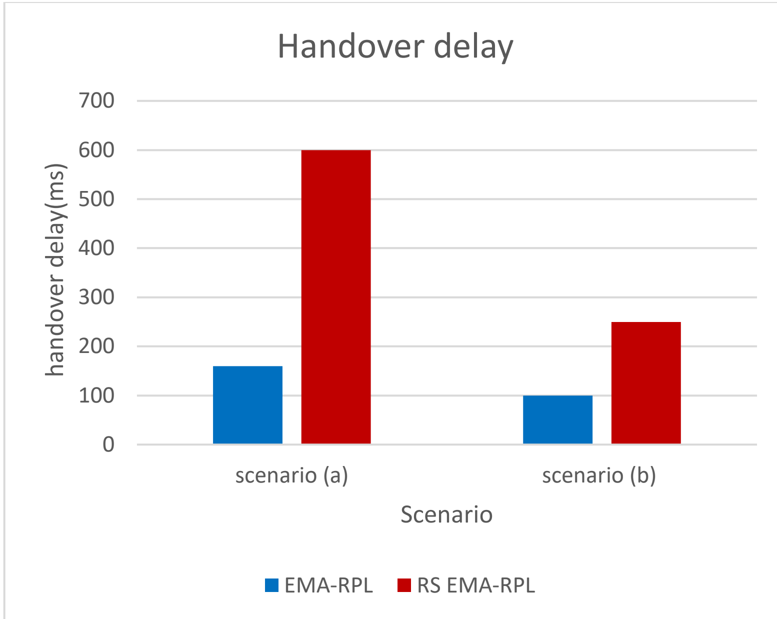

Figure 5.2 Comparison of handover delay

5.2.3 Energy Consumption

The power consumption of the proposed algorithm and the EMA-RPL mobile node are the same. Therefore, here EKFs are compared and the algorithm suggested because both algorithms generate a stable pathway. As can see from the graph in Fig. 6.3, the power exhaustion of the EKF algorithm's mobile node is higher than that of the Energy mobility aware RPL track stable. The power utilization of fixed nodes is approximately identical for both methods. The energy usage is a calculation of various powers like transmission power, message sending power and others. The sum of all these powers gives the total energy consumption in particular unit of time. Generally we avoid high energy consumption of any mobile node as it have limited power resources and the motion already dissipates its energy hence mobile node energy consumption must be least compared to other static nodes.

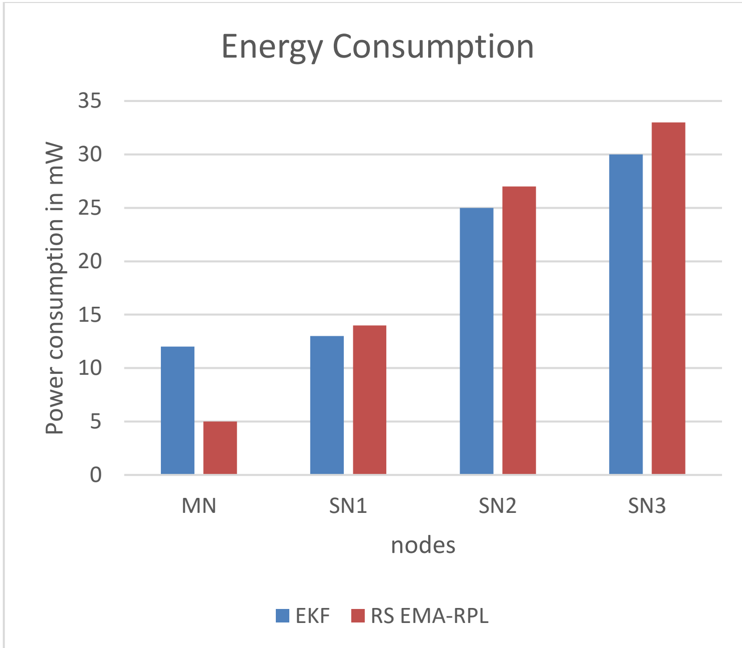

Figure 5.3 Energy Consumption Comparison

CHAPTER 6: CONCLUSION AND FUTURE SCOPE

New algorithm have been proposed to minimize data loss, minimize topology changes, and improve path stability for the EMA-RPL protocol. This approach is considered proactive because the mobile node finds a new parent before separating from the previous parent. Most steps and calculations are performed by the parent node to minimize mobile node resource usage. The proposed protocol can give more precious results if the velocity calculation is carry out by the mobile leaf node on its own, but it can exhaust the assets of the mobile leaf node that require more importance than perfection. Delivery delays are increased, but still lower when compared to the RPL protocol. Reducing delivery delays can be considered for future work and better results can be achieved.

The new proposed protocol also gives better result as compared to EKF algorithm as the mobile node resources utilization is better in the new suggested algorithm and other parameters are almost the same. Even sometimes the route stability of the proposed methodology is better as compared to EKF algorithm.

The new proposed algorithm improves the route stability in EMA-RPL but also increases handover delay which can be a problematic issue in case of some sensitive application like that of health care. A model that can decrease handover delay and increase route stability can be worked upon in future. Beside stability and handover delay there are many more dimensions in which work can be carried out like energy consumption, allowing mobile node to be leaf node and other properties. The route stability increased in the proposed algorithm can improve the battery life of IoT devices which in turn can increase the efficiency of overall network. Data loss is still a big issue that need to be dealt in future as losing important data can be costly for a long terms especially in sensitive applications like healthcare and military. The proposed work can bring a revolution to wireless sensor networks and increase their range in various areas. Beside this new areas can also be discovered in which WSN can be implemented to enhance our day to day life.

References

1. P. López, D. Fernández, A. J. Jara and A. F. Skarmeta, "Survey of Internet of Things Technologies for Clinical Environments," International Conference on Advanced Information Networking and Applications Workshops, Barcelona, July 2013.

2. M. K. Singh, S. I. Amin, S. A. Imam, V. K. Sachan and A. Choudhary, "A Survey of Wireless Sensor Network and its types," 2018 International Conference on Advances in Computing, Communication Control and Networking, pp. 326-330 , July 2019.

3. H. Kim, J. Ko, D. E. Culler and J. Paek, "Challenging the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL): A Survey," IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2502-2525, Sept 2017

4. L.Bartolozzi, F.Chiti, R.Fantacci, T.Pecorella and F.Sgri lli, “Supporting monitoring applications with mobile Wireless Sensor Networks: The eN Route forwarding approach”, IEEE International Conference on Communications pp. 5403-5407, Ottawa, Canada, June 2012.

5. V. Safdar, F. Bashir, Z. Hamid, H. Afzal, and J. Y. Pyun, “A hybrid routing protocol for wireless sensor networks with mobile sinks,” in Proc. 7th International Symposium Wireless Pervasive Computer, Jul. 2012

6. JP Vasseur, Navneet Agarwal, Jonathan Hui, Zach Shelby, Paul Bertrand and Cedric Chauvenet, “RPL: The IP routing protocol designed for low power and lossy networks”, IPSO alliance, April 2011.

7. P. Thubert, “Objective function zero for the routing protocol for lowpower and lossy networks (RPL),” Internet Eng. Task Force, RFC 6552, vol. 20, Mar. 2012.

8. P.Levis, T.Clausen, J.Hui, O.Gnawali, J.Ko, “The Trickle Algorithm”, IETF Request for Comments 6206, March 2011.

9. I. E. Korbi, M. Ben Brahim, C. Adjih and L. A. Saidane, "Mobility Enhanced RPL for Wireless Sensor Networks," International Conference on The Network of the Future, pp. 1-8, Feb 2013.

10. M. Bouaziz, A. Rachedi, “A survey on mobility management protocols in Wireless Sensor Networks based on 6LoWPAN technology”, Elsevier Journal of Computer Communications, Volume: 74, pages: 3–15, Oct 2016.

11. M. Bouaziz, A. Rachedi, A. Belghith,M. Berbineau and S. Al-Ahmadi, “ EMA-RPL: Energy and mobility aware routing for the Internet of Mobile Things”, Future Genereration Computer System, vol.97, pp. 247–258, Feb. 2019.N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses and N. S. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," in IEEE Signal Processing Magazine, vol. 22, no. 4, pp. 54-69, July 2005.

12. Md.S. Hossen, A.F.M.S. Kabir, R.H.Khan and A. Azfar, “Interconnection between 802.15.4 Devices and IPv6: Implications and Existing Approaches”, IJCSI, Vol.7, Iss 1, N.1, Jan 2010.

13. M. Bouaziz, A. Rachedi and A. Belghith, “EKF-MRPL: Advanced mobility support routing protocol for internet of mobile things: Movement prediction approach.”, Future Genereration Computer Syststem, Dec. 2017.

14. O. Gaddour, A. Koubäa, R. Rangarajan, O. Cheikhrouhou, E. Tovar and M. Abid, "Co-RPL: RPL routing for mobile low power wireless sensor networks using Corona mechanism," IEEE International Symposium on Industrial Embedded Systems, pp. 200-209 Aug 2014.

15. Y. Tahir, S. Yang and J. McCann, "BRPL: Backpressure RPL for High-Throughput and Mobile IoTs," IEEE Transactions on Mobile Computing, vol. 17, no. 1, pp. 29-43, 1 Jan. 2018.

16. S. Murali and A. Jamalipour, "Mobility-Aware Energy-Efficient Parent Selection Algorithm for Low Power and Lossy Networks," IEEE Internet of Things Journal, vol. 6, no. 2, pp. 2593-2601, April 2019.

17. S. Min, S. Chung and Y. Ha, "An Improved Mobility Support Mechanism for Downward Traffic in RPL," International Conference on Ubiquitous and Future Networks, pp. 311-313, Aug 2018.

18. M. Barcelo, A. Correa, J. L. Vicario, A. Morell and X. Vilajosana, "Addressing Mobility in RPL With Position Assisted Metrics," IEEE Sensors Journal, vol. 16, no. 7, pp. 2151-2161, April1 2016.

19. H. Fotouhi, D. Moreira, M. Alves, and P.M. Yomsi, “mrpl: Boosting mobility in the Internet of Things”, Ad Hoc Networks, vol.26, pp.17-35, 2015.

20. C. Kiraly, T. Istomin, O. Iova and G. P. Picco, "D-RPL: Overcoming memory limitations in RPL point-to-multipoint routing," IEEE Conference on Local Computer Networks, pp. 157-160, Jan 2016.

21. H. Kharrufa, H. Al-Kashoash and A. H. Kemp, "A Game Theoretic Optimization of RPL for Mobile Internet of Things Applications," IEEE Sensors Journal, vol. 18, no. 6, pp. 2520-2530, March 2018.

22. J. Ko and M. Chang, "MoMoRo: Providing Mobility Support for Low-Power Wireless Applications," in IEEE Systems Journal, vol. 9, no. 2, pp. 585-594, June 2015.

23. F. Gara, L. Ben Saad, R. Ben Ayed and B. Tourancheau, "RPL protocol adapted for healthcare and medical applications," International Wireless Communications and Mobile Computing Conference, pp. 690-695, Aug 2015.

24. F. Gara, L. Ben Saad, E. Ben Hamida, B. Tourancheau and R. Ben Ayed, "An adaptive timer for RPL to handle mobility in wireless sensor networks,"International Wireless Communications and Mobile Computing Conference, pp. 678-683, Sept 2016.

25. Y. Shin and S. Seol, "Improvement of Power Consumption in RPL-based Networks for Mobility Environment", International Conference on Electronics, Information, and Communication, pp. 1-3, April 2020.

26. H. Kim, J. Youn, H. Kim, S. Yoon and S. Bahk, "Demo: Mobility Enhanced RPL for General Mobility Scenarios," IEEE Wireless Communications and Networking Conference Workshops pp. 1-2, June 2020

27. S. Hoghooghi and R. Javidan, "Proposing a new method for improving RPL to support mobility in the Internet of things", IET Networks, vol. 9, no. 2, pp. 48-55, March 2020.

28. A. Kumar and N. Hariharan, "Enhanced Mobility Based Content Centric Routing In RPL for Low Power Lossy Networks in Internet of Vehicles," International Conference on Intelligent Autonomous Systems (ICoIAS), pp. 88-92, April 2020.

List of Publications

1. R. K. Yadav and N. Awasthi, "A Route Stable Energy and Mobility aware routing protocol for IoT," 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2020, pp. 942-948.

2. Paper titled “A survey on enhanced RPL: Addressing the mobility in RPL”, accepted in 4th International conference on I-SMAC (IoT in social, mobile, analytics and cloud).