# Enabling Green Video Streaming over Internet of Things

($3^{rd}$ Quarter Deliverable)

Dr. Ghalib Asadullah Shah (PI)

Next-generation Wireless Networking (NWN) Lab,
Al-Khawarizmi Institute of Computer Sciences,
University of Engineering and Technology, Lahore

12-09-2014

# Contents

# List of Figures

# List of Tables

# About this Document

This document reports the activities performed in the 3rd quarter of our project 'Enabling Green Video Streaming over Internet of Things' and the corresponding deliverable to be submitted to ICT R & D Fund. In the first deliverable we conducted a detailed literature survey of IoT communication protocols and identified the issues for IoM which are to be addressed in this project. The most significant task was to propose the suitable link layer, which is IEEE 802.11, for IoM devices considering the low data rate of IEEE 802.15.4 implemented in IoT communication stack. Later in the second deliverable we conducted a detailed performance analysis of IEEE 802.11 in terms of energy efficiency for its suitability in multimedia sensor devices. The analysis spanned on the IEEE standards proposed power saving techniques. The outcomes of this analysis suggested a great deal of potential in conserving energy in IEEE 802.11 in order to realize in IoM.

In this deliverable, our contributions are two-fold; First, we designed an IoM architecture to realize multimedia communication as a specialized subset of the IoT. The proposed IoM architecture is first of its kind and to the best of our knowledge previously no architecture of IoM has been presented in literature. The architecture specifies stringent requirements of the IoM and the possible technologies that can be adopted in the IoM communication stack. In the second part of this deliverable, we present a green communication analysis of a light weight routing protocol RPL that is proposed for the communication stack of IoT by the IETF ROLL working group. RPL routing protocol is a highly flexible that allows us to conduct its energy consumption analysis and measure its performance in terms of the energy efficiency and network performance. In the literature, no research work is available in which the performance analysis of three objective functions i.e Hop count, ETX, and Energy, is done simultaneously. We explore the flexibility of the RPL and observed its potential to be used as a green communication protocol. Literature survey suggests that RPL has been used as an energy efficient routing protocol. Moreover, no prior study has investigated the potential for low carbon footprint and green communication in routing functions. We will address them in our proposed protocol, which will be developed in the next deliverable.

On the basis of the findings of this deliverable, we will design an energy efficient and green routing protocol in next deliverables that will be tuned to support real-time and green multimedia communication as proposed in this project.

# Chapter 1

# IoM Architecture Design

The introduction of Internet of Things (IoT) has triggered the need to realize Internet of Multimedia (IoM) in order to truly support the notion of things connectivity to the Internet. The current task group activities in IoT does not mandate the features of multimedia objects, thus leaving a gap to benefit from this paradigm for developing multimedia applications. In this report, we present the vision of IoT and draw the inspiration towards the perspective vision of IoM. Therein, we introduce IoM as a novel paradigm in which each multimedia device is connected by a unique IP address and can be uniquely accessible and able to deliver its contents to the requesting user or server across the Internet.

## 1.1 Introduction

Physical objects or things, equipped with the capability to observe and/or interact with physical environment and the ability to communicate with other things, are extending the Internet towards the so called Future-Internet, now commonly referred to as 'Internet of Things' (IoT) [1, 2, 3]. IoT comes with new challenges due to various degrees of resource constraints of the devices, heterogeneous classes of acquired and shared data and the enormous scale of connectivity. Challenges posed by IoT are not limited to the communications side of the equation. Supporting technologies like sensors, actuators, RFID devices, etc and the processing capabilities based on pervasive computing, ambient intelligence, context aware computing, etc are also required to realize Internet of smart things [4, 5]. Consequently, IoT calls for revisiting the communication and computational technologies in the context of "smart things" being part of the global network.

Existing surveys on IoT [1, 2, 3] define it in terms of the things, Internet, enabling technologies, exploring the market potential of these technologies, contemplate the hurdles towards standardization and wider implications of the IoT for society i.e. growing concerns over privacy. However, none of these surveys consider the challenges posed by multimedia devices or the transportation of multimedia traffic over the network along with other scalar data. Inherent characteristics of multimedia information impose a number of restrictions on the design of IoT, in addition to the challenges imposed by other devices which are part of the IoT. To meet given quality of service (QoS) requirements, network characteristics such as bandwidth, end-to-end delay, jitter and error rate are required to be regulated to ensure acceptable delivery of the multimedia content.

Multimedia enabled IoT, or IoM as it will subsequently be referred to, promises a wide array of applications in both commercial and military domains. Popular IoT applications such as smart homes, smart cities, distributed surveillance systems, traffic monitoring and control, industrial process control, environmental and oceanographic monitoring will be more effective by incorporating multimedia information.

Prior studies on multimedia communication over wireless networks have been conducted with a limited scope based on specific types of devices, applications, content, among others. However, none of these studies focus on heterogeneity of multimedia devices and global networking issues that will arise in realizing IoM. Thus, there is a need to develop a new paradigm that focuses on these issues collectively. In this article, we provide the concept, vision and applications of IoM and analyze its distinct characteristics as compared to the already employed multimedia networks in terms of architecture and requirements. Apart from the potential enabling technologies, we present the technical specifications and requirements for the realization of IoM. We also discuss the communication protocols designed for IoT and analyze their feasibility for IoM. In addition, the potential multimedia processing technologies are presented that can facilitate efficient multimedia communication, specifically via wireless multimedia device. Thereby, we identify requirements and challenges specific to IoM. We aim to promote the vision of IoM and identify its stringent requirements and challenges which makes it a special subset of IoT, that needs further consideration to practically realize the tremendous number of IoM based services and applications.

## 1.2 Proposed IoM Architecture

In Internet of Multimedia (IoM), services and applications are based on the information that is essentially provided by the multimedia devices. Like conventional multimedia networks the applications based on IoM require stringent QoS requirements to provide the satisfactory user-experience. Similarly, the some of the applications of IoM may be loss-tolerant, on the other hand some multimedia applications require delay-sensitive communication.

In conventional multimedia networks, the end-nodes (multimedia devices) just report the acquired multimedia information from their vicinity and this multimedia content may be streamed at the destination node or stored in a storage hub (server) for later use [6, 7, 8]. Generally, these multimedia devices are designed to be able to communicate to other multimedia devices possessing similar character-

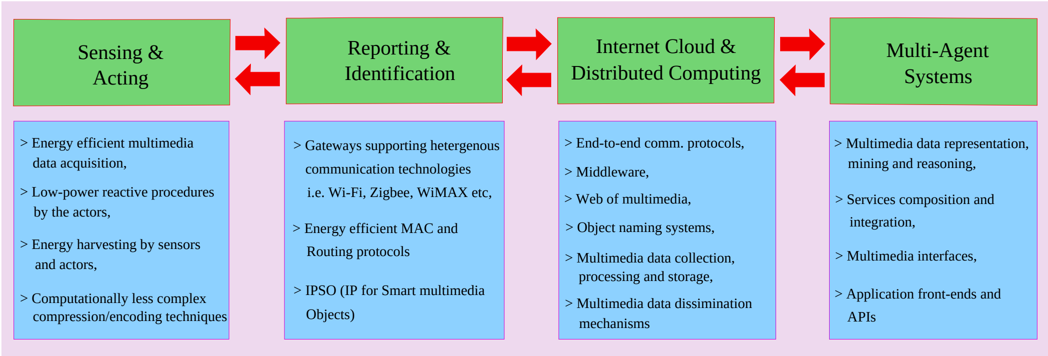


Figure 1.1: IoM Architecture

istics (homogenous devices) i.e. similar communication stacks, similar resources, etc. However, IoM envisions to enable each multimedia device i.e. cameras, microphones, etc, to be globally accessible by a unique IP address with the same spirit as of the computers and other networking devices connected through the Internet. Moreover, some computing capability can be embedded in the multimedia devices to make them smart enough to perceive the system and service requirements and trigger actions on their own. Thereby, heterogeneous multimedia devices acquiring multimedia content from the physical environment i.e. audio, video, images, etc, can communicate and interact with each other as well as to other 'smart objects or things' connected in the global network cloud via the Internet. This approach enables a wide range of applications in the areas of home and building automation, factory monitoring, smart cities, transportation, smart grid and energy management [9].

Recently, due to the availability of low-cost low-power multimedia devices i.e. Complementary Metal Oxide Semiconductor (CMOS) cameras, CMOS-MEMS microphones, etc have gained lots of attraction in wireless multimedia networks. As it is said that, 'a picture is worth a thousand words', the multimedia data provides very comprehensive information which can be adequately standardized in appropriate formats, models, semantic description based on the context to realize it as a useful information or knowledge. Consequently, it can be expected that the IoM has the potential of enormous number of applications and it will be an essential part of the IoT. However, the incorporation of radically heterogeneous multimedia devices forming a highly dynamic and large network, admitting non-multimedia devices as well, necessitate to reinvestigate the currently employed architectural designs and communication procedures followed in the Internet. Thereby, the vision IoM is depicted in a four layered architecture in Fig. 1.1. The bottom layer labelled as *Sensing & Acting* comprises of the devices which are sensing and reporting data to the data gathering and processing server. The server in turn may generate some commands and actuators runs those commands. This layer basically interacts with the physical environment. The second layer known as *Reporting &*

*Identification* deals with the networking and communication service for interaction between the devices and the data processing server. Considering the scale of the IoM devices, their future growth and enormous applications, the architecture proposes an *Internet cloud & distributed computing* layer for data processing and enabling various services. Finally, the top layer provides an interface and agents for application development and user interaction with the IoM. The proposed architecture put forward the unique requirements and challenges in realizing multimedia communication over Internet of things or simply Internet of Multimedia. Each layer in the architecture represent distinct phases of multimedia data sensing, communication, and processing, which enables multimedia based services and applications. The following section describes the four layers in detail.

### 1.2.1 Sensing & Acting

Unlike the scaler data acquisition by the sensor nodes in IoT, the multimedia data acquisition from the physical environment is huge in size. Small sized multimedia devices have limited memory resources, thus this data needs to be quickly processed and transmitted on the air to vacant the space in the memory for the incoming data as the multimedia data is continuous in nature. Similarly, the acquired huge amount of multimedia data undergo various pre-transmission processing procedures at the multimedia device i.e. transformation, quantization, estimation, entropy coding, etc, so that it can be compressed to reduce bandwidth requirements while transmission. These processes are computationally complex and consume significant amount of energy. However, various promising solutions proposed for efficient multimedia communication like compressive sensing [10], distributed video coding [11], etc can be utilized to facilitate multimedia acquisition and processing at the multimedia device in IoM.

The recent advancement in designing low-cost and small scale devices using the enabling technologies such as MEMS, have promoted to embed electronic devices in small objects [12, 13]. This development will enable Internet connectivity for typical machines around us like cars, fridge, lights, etc, as well as for things from our everyday life like our shoes, cornflakes box in our fridge, camera on our front door, etc. In addition, some basic computing intelligence is added in smart devices to communicate and interact with each other and with the physical environment making them smart objects or things that is based on open standards to build an IoT. Based on the multimedia data acquired by the sensor devices, these devices can initiate some reaction procedures which can be performed on the same multimedia device or at some other smart thing or object.

In most of the cases the multimedia devices in IoM are expected to be battery powered. Since, the multimedia acquisition and its processing are very power consuming procedures. Thus, in IoM the multimedia devices should harvest energy from different energy sources. Four commonly used energy harvesting means are as follows: (i) Photovoltaic; in indoor (i.e. in an office) as well as outdoor environment (i.e. sunlight), (ii) Vibration; movement of cars on a bridge with sensor on the bridge, (iii) Thermal; human body heat at ambient air, (iv) Radio Frequency; power harvesting from GSM signals in a city. To prolong network lifetime equipping multimedia devices with multiple energy sources i.e. solar cells, will result in a substantial increase in useability of IoM.

The real-time multimedia content is continuous in nature which may or may not be delay tolerant depending upon the given audio/video application i.e. streaming or interacting. For example, a video camera device generates 25 to 30 frames per second from the optical data. These frames are transmitted in the form of packets requiring specific bandwidth capacity directly related to video compression being done at the transmitting device. Compressing video to enable a certain level of quality over a low capacity channel offering low data rate, increases encoder complexity and energy consumption. On one side energy constraint IoM device exhibit limited bandwidth capacity, yet enabling a good video quality requires high compression which is also infeasible due to high energy consumption. Thus, there is a tradeoff between the achievable compression and the energy utilization for a specific level of user experience restriction.

### 1.2.2 Reporting & Identification

The multimedia content communication over the network impose stringent traffic management requirements as compared to data communication, in terms of the bandwidth, delay, jitter, reliability, etc [7, 8]. These network performance requirements are referred as the QoS, which represents the level of user experience needed for a specific application. For example, how fast is the data transmission? how much is the delay at the receiver? what is the probability of correct data reception at the receiver? what is the probability of the transmitted data to be lost? among others.

IoM based networks are expected to be large in scale with huge number of devices. Obviously, these devices will be equipped with different communication and computational capabilities due to the their heterogeneous resources and operations [14, 3, 5]. Therefore, to enable communication between devices with heterogeneous technologies the gateway nodes should be equipped with multiple interfaces in order to support multiple communication technologies i.e. WI-Fi, WiMAX, ZigBee, etc.

The current standardization activities of providing Internet access to things [15] are not focused to address the challenges of provisioning multimedia objects over IoTs. Things in IoT are commonly assumed to be small objects generating periodical or event driven scalar data in the order of few tens of octets. Although ZigBee [16] or IEEE 802.15.4 standard [17] (currently proposed for IoT) is devised for low-power devices to operate at short range for IoT, it is not feasible to employ it for wireless multimedia sensors, present in IoM, due to its low rate in the range of 50-250 kbps. However, the most widely video encoding standard used for small embedded devices is H.264 that typically produces the Bit-Rate even at low resolution (384 kbps at 320x240 at 20 frames per second) much higher than the maximum data rate of 250kbps in IEEE 802.15.4. Moreover, the video frame size is in the order of kilo-bytes that would require a large number of physical layer frame transmissions and eventually incurring much higher overhead that cannot be efficiently used for multimedia delivery. Thus, it will prevent the use of multimedia devices to connect through IoT defined protocols and specification. Therefore, multimedia traffic over IoM requires a decent data rate supporting communication technology like IEEE 802.11, Bluetooth version 3 and 4, etc

IoT aims to enable every device or thing to be accessed or connected to Internet for a variety of reasons. The IP for Smart Objects (IPSO) Alliance promote the Internet Protocol as the network technology for connecting Smart Objects around the world with the help of IPv6 [18]. Through IP adaptation and incorporation by low power network standards like IEEE 802.15.4 will enable deployment of IoT paradigm [18], but the latency constraint of multimedia devices have been overlooked in this effort. Thus, enabling small multimedia sensor devices to efficiently communicate with heterogeneous devices is still need to be devised.

Resource constraint multimedia devices require implementation specific QoS, which is realizable with network metrics i.e. delay, latency, jitter, etc. Thus, IoM requires communication standards such as HTTP [19], IP [20], TCP [21], etc. However, direct implication of these protocols for IoM is not straight forward, since these protocols are not designed considering devices having energy constraints. On the contrary, IoM multimedia devices are realized as low-cost, battery powered, low-complexity devices and the existing protocols i.e. HTTP, IP, TCP, consume significant amount of energy due the redundant data transmissions, protocol overheads, headers transmission, acknowledgement packets for higher layers to ensure reliability, etc. Since, these communication protocols are not optimized for low power communication as required in IoM, therefore adaptation of these mechanisms is infeasible in their current form and structure.

### 1.2.3 Internet Cloud & Distributed Computing

Mostly, the network devices that interact with physical environment are embedded with minimal intelligence and their major responsibility is to report the statistics or other information to a centralized processing center or directly relay the information to the end-user. The response strategies and actions are generated by the centralized processing systems or the end-users. Generally, the communication networks are based on homogeneous proprietary and non-proprietary solutions, and these systems can be realized as connectivity islands, and they require some application definite gateways to export data through Internet to the destination nodes. Also, the simple network devices interacting with physical realm i.e. sensors, actuators, etc, are not directly addressable, therefore the inter-operability between the heterogeneous systems realizing them as a single large virtual systems requires hardware as well as application compliant solutions.

In IoM the multimedia data is not only required to be delivered to the end-user using end-to-end communication protocol, in addition the user should be able to initiate various supported operations on the smart devices. For example, a user may want to see the multimedia stream from a particular device or user may want a particular device to start or stop its operation, etc. To make multimedia information coming from number of sources more understandable and illustrative, web of multimedia might be used. Web of multimedia can enable a user friendly representation of the multimedia sensing data to the user. Similarly, the multimedia devices can be distinguished on the bases of their IP addresses. However, IP addresses are understandable to machines and were never intended for humans to remember them. In an object naming system (ONS), the smart things or objects are given name and to communicate with a particular thing its name is used by human.

The multimedia data require enormous amount of space (memory). Thus, in IoM architecture it is proposed to store it on a cloud. The cloud has the capability to store huge amount of data as well as the high processing capability. Thereby, various computing and processing technologies like pervasive computing, ubiquitous computing, service-oriented computing, cloud computing, etc provide architectures and functionalities to enable various applications and services on top of the deployed networks. Using middleware the application developers can access data from the cloud and build applications on top of it. The cloud is responsible for dissemination of multimedia data to the users and process the operations on the data as specified the application.

### 1.2.4 Multi-Agent Systems

The huge amount of multimedia and other sensing data at the cloud needs to be given useful representation by using data mining and reasoning techniques. Similarly, using distributed artificial intelligence multiple software which are usually refereed as 'agents', are enabled to interact with each other and with the environment to facilitate the required services and applications. Different services can be composed on the same big data and multiple services can also be combined to assist another global service or application. In addition, multimedia interfaces have to be provided for the end-users and application developers. Similarly, the APIs to huge amount of data from various number of sources as well as access to the multimedia sensing devices should be provided for application developers to build targeted applications.

## 1.3 Project Scope

The scope of this project is limited to the functions of lower two layers, i.e., data acquisition and transmission. However, some of the functions of higher layers will also be implemented to build and demonstrate IoM architecture. Perhaps, a new proposal will be submitted to address the entire challenges of higher two layers.

The major challenges and requirements for IoM that will be addressed in this project are as follows:

- Energy efficient multimedia content acquisition from the physical environment to assist longer network lifetime.
- Efficient encoder design to compress multimedia data, so that transmission overhead is minimized.
- Low-complex encoder design that is feasible for resource constraint devices having low-processing capability and limited energy resources.
- Energy efficient routing protocol to facilitate delay intolerant real-time multimedia communication, green communication and long network lifetime.
- Energy efficient MAC layer protocol to facilitate efficient channel access mechanism, provide high data rate to enable high quality multimedia transmission and support long network lifetime.
- Adoption of multiple energy sources to harvest energy from environment in addition to the charged batteries, so that network lifetime increases.

# Chapter 2

# Performance of Existing Green Communication

In large scale wireless multimedia networks, the data is communicated to destination nodes by multiple hops in between wireless links i.e. multi-hop communication. To forward a data packet from source to destination a path towards the destination should be known to the source node, this path is determined by routing protocols. There are two types of routing protocols; (i) Reactive routing protocol, in which path towards a destination is found only when it is required. (ii) Proactive routing protocol, in which a path from every source to every destination is maintained at the start of the network whether it is required or not. Some popular reactive and proactive routing protocols are AODV, DSR and OLSR, DSDV, respectively. The operation of these protocols is independent of the dynamics of application requirements, that is why they are ill suited for IoM.

In IoM the wireless multimedia sensor nodes are required to be capable of communicating with one another and to the gateway node. In addition, to support real-time multimedia communication the overhead of finding a path in terms of time is critical. Therefore, a proactive routing protocol is preferred. As real-time multimedia communication are delay intolerant, thus routing protocol should also ensure to select paths with lower end-to-end delays. Similarly, another aspect of routing protocol for IoM is its support to ensure green communication. The green communication has two concepts, one is to minimize energy consumption and second is to minimize carbon footprints.

IETF routing over low power and lossy network (ROLL) group has proposed an IPv6 proactive routing protocol for IoT named Routing protocol for low power and lossy networks (RPL). This protocol is light weight in terms of energy consumption and memory requirements, which makes it very suitable for resource constraint multimedia sensor nodes in IoM. In addition, control packet format and its frequency of transmission is controllable to save energy and support node metrics sharing. Unlike conventional routing protocols the RPL supports multiple node metrics sharing capability, these metrics are used to maximize or minimize a specific objective function. For example, if nodes share their battery status in control packets then a path with longer lifetime can be selected, resulting in stable network operation. Due to this flexible operation of RPL, we propose it for adoption in IoM communication stack. RPL can support multiple objective functions for multimedia traffic such as delay of the path, a path with least carbon footprints, a path with high reliability, etc. For this reason the RPL routing protocol has the potential to adapt to the dynamic applications of IoM due to its flexibility.

## 2.1 Routing Protocol for Low power and Lossy networks (RPL)

RPL forms a directed acyclic graph (DAG) which maintains a tree like topology. Each node in a RPL network has a preferred parent which acts like a gateway for that node. If a node does not have an entry in its routing table for a packet, the node simply forwards it to its preferred parent and so on until it either reaches the destination or a common parent which forwards it down the tree towards the destination. The nodes in a RPL network have routes for all the nodes down the tree. It means the nodes nearer to the root node have larger routing tables. Path selection is an important factor for RPL and unlike traditional networks routing protocols, RPL uses more factors while computing best paths for example routing metrics, objective functions and routing constraints.

RPL uses TCP/IP for communication of the control packets among the network nodes. In this way IP is used in LLN to provide end-to-end connectivity, which enables communication between heterogenous (multimedia) devices. Experience has shown that IP is both lightweight enough to run on even severely resource constrained systems. In the following sections we describe RPL topology, routing mechanism, RPL control messages, routing metrics, constraints, Objective functions, Trickle timer and finally an example about RPL working.

### 2.1.1 Network Topology

The network topology is not fixed in wireless networks that is why RPL forms a topology in a tree like graph. In this tree the root node is at the top and the leaf nodes are at the edges connected to the root node through intermediate parent nodes. Unlike traditional tree like topology the RPL routing protocol provides the capability to maintain redundant roots towards the root node. Thus, it can mitigate the lossy nature of the wireless channel. As RPL uses DAGs which maintain directed routes, thus both upward and downward routes are maintained to enable traffic flowing in both directions. The upward means the uplink traffic from the leaves nodes to the root and downward traffic means the traffic flow from root to the leaves node (downlink traffic). A typical RPL topology is shown in

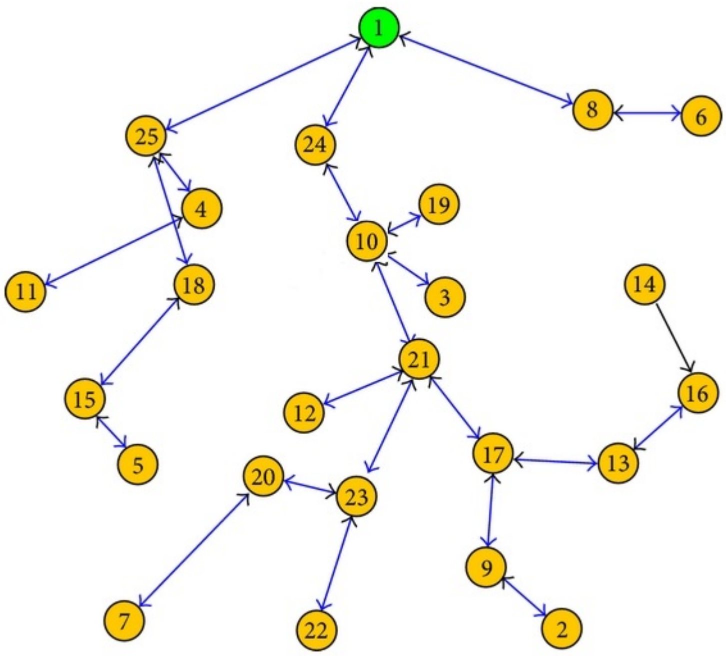


Figure 2.1: RPL Topology

Fig. 2.1.

### 2.1.2 Routing in RPL

RPL maintain the topology information in a Destination Oriented Directed Acyclic Graph (DODAG). The DODAG contains the paths from the leaves to the root, which is also referred as border router or gateway node. When a leaf node wants to send traffic to the root node, it simply sends the packet to its preferred parent which is already chosen by it using the DODAG it received. Its preferred parent in the tree then sends the packet to its preferred parent and so on until the packet reaches the root. To avoid loops in the route, a rank is assigned to every node in the network. The root always has rank 0 and nodes closest to the root is assigned rank 1 and next node gets rank 2 and so on i.e. every child node has higher rank than its parents.

RPL uses three types of control messages for creating and maintaining RPL topology and routing table. (i) DODAG Information Object (DIO) messages are used by RPL to form, maintain and discover the DODAG. When a RPL network starts, the nodes start exchanging the information about the DODAG using DIO messages. (ii) DODAG Information Solicitations (DIS) is used by any node to explicitly solicit the DIO messages from the neighbor nodes. It is triggered by the node in case when it could not receive a DIO after a predefined time interval. (iii)

DODAG Destination Advertisement Object (DAO) messages are used by RPL to propagate a node prefix to the ancestor nodes in support of downward traffic.

A RPL instance identified by RPLinstanceID may contain several DODAGs identified by DODAGID. Different DODAGs are necessary for steering different types of network traffic. Each DODAG has its own OF, metrics and sink. The DODAG uses DODAGSequenceNumber to show the freshness of the information. The nodes nearby will receive the DIO from the root and will process it as it is from a lower rank node and will select root as their parent. These nodes will now send link local multicast DIOs and the other nodes receiving the DIO may select them as parent. If a node receives DIOs from two or more parents, it will decide based on the OF (e.g. optimize path ETX, or prune battery operated nodes etc). This process will continue until all the nodes join the DODAG. Once the topology is formed using the DIO messages, then for downward traffic flow downlink paths are maintained using DAO packets. DAO packets are sent by the children to their parents and so on, up to the root node. In this way intermediate as well as the root node determines the paths for downlink communication.

### 2.1.3 Routing Metrics

In multihop wireless networks multiple paths can exist to a single destination. To chose the best path routing metric is used. Routing metric values quantitatively represent the cost of a path. In RPL path is chosen according to the objective function based on some routing metrics. The choice of the routing metric is very important to facilitate the application running on the application layer such as if the application requires traffic transmission in a given delay bound then link quality along the whole path is the routing metric. Similarly, in case energy efficiency and green communication is intended then the routes with high level of remaining energy and equipped with green energy sources are preferred, thus battery status and energy source should be the routing metrics here.

In addition to the conventionally used link metrics like Throughput, Latency, Link Quality Level, ETX, Link Color, the RPL routing protocol also supports node metrics i.e. Node State Attribute (NSA), Node Energy, Hop count. Moreover, a routing constraint is used to either include or exclude links from the routing path that do not meet the criteria specified in the objective function.

### 2.1.4 Objective Functions

When a child node receives multiple DIO packets from its neighboring parent node, it applies the metrics received to calculate the path cost through each of the possible parents according to an objective function (OF). The objective function is minimized or maximized for a certain parent that is specific to the application needs and that parent is selected as the preferred parent.

Two popular objective functions that are implemented in multiple operating systems designed for IoT are OF0 and ETX. OF0 uses hop count as routing metric where as ETX uses expected number of transmissions metric as a routing metric for selecting the best path. OF0 ensures that minimum number of intermediate nodes are used within a path towards the root node. On the other hand ETX minimizes the number of time a single packet is transmitted before it is received by the root node.

### 2.1.5 Trickle Timer

The formation of network topology and then updating the route changes required transmission of control packets which needs to be minimized to save energy. The most significant overhead among the control packet of the RPL is the transmission of DIO. Fortunately, the RPL routing protocol uses a trickle timer to control the overhead posed by the DIO control packets. The smallest possible interval between two DIOs is equal to DIO Minimum Interval which keeps on increasing (doubling) until it reaches the maximum value determined by DIO Interval Doublings. There are two main controlling parameters for the trickle timer, Imin and Imax. The Imin parameter gives the minimum amount of time between two DIOs. So, the minimum value of the trickle timer is Imin. This value then doubles to a reach the maximum value of the interval between two DIOs that is Imax.

## 2.2 Related Work

Low-power and lossy link networks require efficient routing of data traffic as well as efficient use of constraint resources. Due to the high flexibility of RPL which makes it suitable for the case when the same network has to support multiple application dynamically. A number of studies have been done to optimize the performance of RPL in various network conditions and requirements. Since, IETF ROLL working group has not provided any standard parameter settings and routing metrics, thus RPL metrics need to be fine tuned as per the specific application requirements.

Energy efficient operation of the routing protocol strictly depends upon the control traffic that is exchanges to find and maintain the routes as well as the choice of paths being selected. RPL uses trickle timer to control the number of control packets being sent by each node [22]. When a RPL network is initiated the control traffic overhead is relatively high, however it decreases once the network routes are stabilized [23]. In [24] it is reported that in a network of 20 nodes operating under a packet error rate of 1%, the control traffic overhead oscillates around the 25%. However, in [24] it is reported that the control overhead increases up to 75% for 100 network nodes. This overhead needs to be reduced to save energy by calibrating the RPL parameters like trickle timer.

The energy routing metric if used alone in the objective function then it may result in high packet loss ratio [25]. Therefore in order to increase the lifetime of the network as well as the efficient packet delivery ratio the energy metric of nodes as well as the link quality metric should be used both in the objective function to get energy efficient network performance. In [26] a duty cycle aware routing scheme is proposed in which instead of addressing the packet to particular intermediate nodes, the packets are forward to all potential receivers and the node that wakes up earliest forwards the packet. In this way the packet delays and energy consumption is significantly reduces as well as the duty cycle of the network nodes.

Not every node in IoT networks have same role in terms of sensing the required data or forwarding the traffic. Therefore, in [27] a research-oriented energy efficient scheme is proposed for RPL routing protocol. The proposed scheme uses multiple metrics such as the amount of energy utilized by a node, factors responsible for the battery consumption which defines how much a node is prone to energy consumption ? i.e. a node closer to root has to forward the traffic of other nodes. In addition, a research oriented metric is introduced only those nodes which support the required sensing data are chosen for routing. The proposed protocol is reported to save significant amount of energy without compromising on the throughput. Similarly, in [28] the energy metric is used to find the cost effective path and then transmit power is decreased using probing technique to save additional energy.

All the studies stated above specially for energy metric based objective function lacks the evaluation of RPL with respect to tweaking its own parameters for instance DIO interval, Duty Cycling interval, etc which plays a vital role in network convergence, Control Traffic overhead, and Energy Consumption. Similarly, their have been no consideration given whatsoever about the carbon footprint emission. Therefore, the energy metric based objective function performance evaluation and green routing protocols for IoT are still open issues.

## 2.3 Evaluation Platform

We have evaluated the performance of RPL for IoM by adding a new objective function of *energy consumption* in Cooja simulator. The Cooja simulator is developed along with the Contiki operating system designed specifically for Internet of Things.

### 2.3.1 Contiki-OS

Contiki is a wireless sensor network operating system (OS) and consists of the kernel, libraries, the program loader, and a set of processes [29]. It is the most popular operating system for IoT things and it has been vastly used by the research community for simulation and real-implementation of IoT based wireless networks.

Contiki provides mechanisms that assist in programming the smart object applications. It provides libraries for memory allocation, linked list manipulation and communication abstractions. It is developed in C, all its applications are also developed in C programming language, and therefore it is highly portable to different architectures like Texas Instruments MSP430.

The Contiki operating system provides modules for different tasks (layers). In addition the functionality of the routing protocol is divided into multiple files each performing one of the major tasks of RPL such as file named rpl-dag.c is responsible for creating and interpreting the DODAG frames. This feature makes it very easier to program different objective functions and tuning the values of key parameters to modify the routing protocols performance.

### 2.3.2 Cooja-Simulator

Cooja is a Java-based simulator designed for simulating sensor networks running the Contiki sensor network operating system. The simulator is implemented in Java but allows sensor node software to be written in C. In Cooja all the interactions with the simulated nodes are performed via plugins like Simulation Visualizer, Timeline, and Radio logger. It stores the simulation in an xml file with extension 'csc' (Cooja simulation configuration). This file contains information about the simulation environment, plugins, the nodes and its positions, random seed and radio medium etc.

## 2.4 Results

To evaluate RPL performance, we consider three routing metrics based objective functions in RPL, i.e. hop count, expected transmission count (ETX) and energy.. We simulated a network comprising of 21 nodes, in which 20 nodes are sensor nodes sensing data and transmitting to a root node. The sensor nodes periodically transmit UDP packets of 26 bytes at an interval 0.8 secs. The sensor nodes also use radio duty cycling (RDC) to minimize energy consumption and check the channel 8 times in a sec i.e. 0.125 secs.

In RPL, when a child node receives a DIO packet from its neighboring parent nodes, it has to chose the preferred parent towards the root node. Among the possible parent nodes the preferred parent is the one on which the objective function gives the smallest path cost. If hop count is used then child chooses the parent through which root node can be accessed using minimum number of intermediate hops. Hop count based objective function is expected to find the shortest path. However, it may not necessarily be the quickest path since the quality of the channel is not considered. Thus, it may result in high number of retransmissions which results in high energy consumption and lower network lifetime.

The ETX metric calculates the expected retransmission count on all the links between the parent and the root node. Thus, among the possible parents the parent with smallest ETX path cost is chosen as the preferred parent. This metric considers the link conditions in the objective function that is why it is considered to be the highly energy efficient metric. Also, due to its energy efficiency the network lifetime increases. However, it may incorporate longer paths in terms of the number of intermediate nodes. The third popular metric is the node remaining energy, in energy based objective function the parent node advertises the status of the minimum amount of remaining energy in its path towards the root in the DIO packets. The child node chose the parent which advertises the highest minimum remaining energy value. This metric tries to ensure fairness of energy usage among nodes. This metric also expects to increase the network lifetime. However, the link conditions are not considered in the objective function that is why the number of retransmissions may increase resulting in high energy consumption.

The default values set for DIO minimum interval is 4 secs = $2^{I_{min}}$ (default value of $I_{min}$ is 12), which is doubled every interval till it reaches the DIO maximum interval that is set to 1024 sec = $DIOminimuminterval * s^{I_{max}}$ (default value of $I_{max}$ is 8). The packet reception ratio is set to 100% i.e. every transmitted packet can be successfully received within the transmission range. The simulation is run for 30 mins and the results are computed by averaging the them over 3 separate simulation runs. Network performance metrics considered are average energy consumption by a network node, control traffic overhead and average packet delivery ratio. The RPL and network parameters that we tune to observe the performance of RPL are DIO maximum interval, DIO minimum interval, Radio Duty Cycling (RDC), and Packet Reception Ratio (PRR).

### 2.4.1 Energy Consumption

We evaluate the per node average energy consumption when the DIO maximum interval parameter $I_{max}$ is changed from three to 15. As shown in Fig. 2.2 that the larger DIO maximum interval is, the lower the energy consumption is observed. It is due to the fact of longer time interval between DIO transmissions. However, when DIO minimum interval parameter $I_{min}$ is changed from three to 15 then the results dramatically change. As shown in Fig. 2.3 that the larger DIO minimum interval values has significantly reduced the energy consumption. Although higher values are preferred but if we further increase the values then energy consumption increases. There is a range of values for which we can achieve minimal energy consumption i.e. Imin = 7 to 9.

The effect of RDC is negligible on the energy consumption since the traffic generation rate is higher, which does not permit to sleep for energy saving. However, as shown in the Fig. 2.4 that ETX achieves more energy efficiency compared to other metrics because it prefers better quality link avoiding retransmission. Next we varied the PRR that is the packet loss rate. As shown in Fig. 2.5 that the is not much difference among the three routing metrics but still ETX performs better.

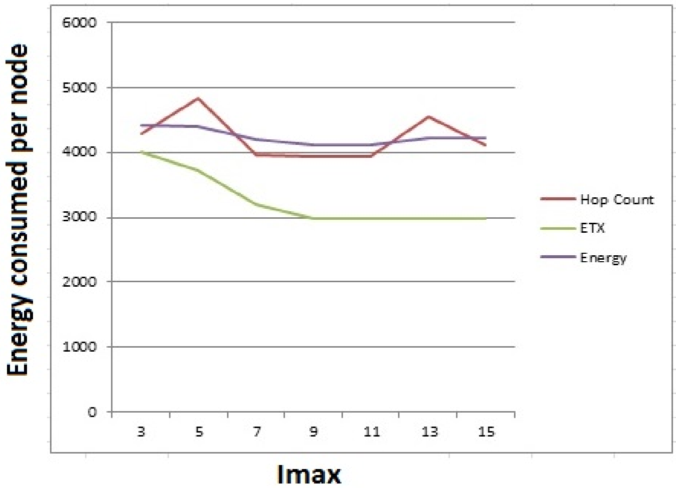


Figure 2.2: DIO MAX vs Energy Consumption

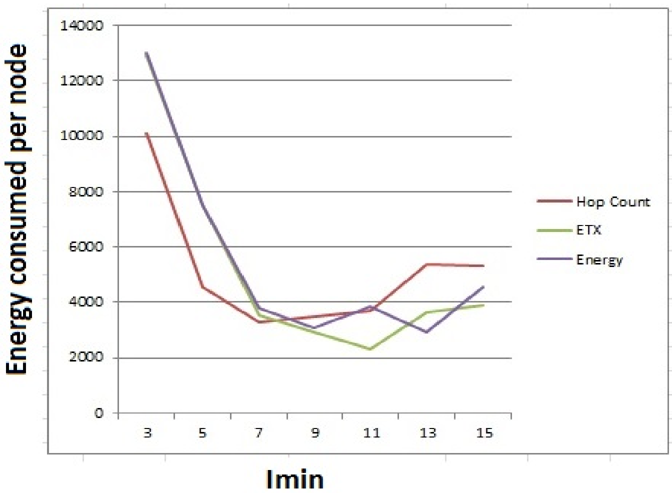


Figure 2.3: DIO MIN vs Energy Consumption

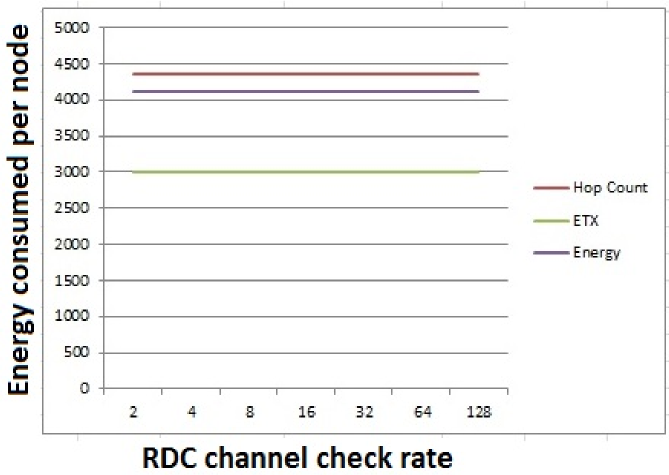


Figure 2.4: RDC vs Energy Consumption

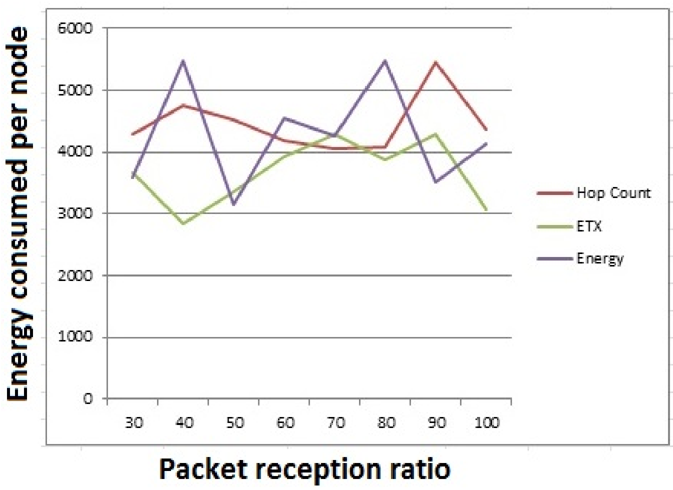


Figure 2.5: PRR vs Energy Consumption

### 2.4.2 Control Traffic Overhead

The control traffic overhead strictly depends upon the trickle timer values. Therefore, when we varied the values of DIO maximum and DIO minimum intervals the control overhead significantly reduces for larger values. As shown in the Fig. 2.6 and 2.7 the number of control packets i.e. DIO, DIS, and DAO, notably decrease when DIO minimum or DIO maximum value is increased. The three routing metrics have also similar performance as compared to one another. However, in case of RDC the metrics performance is different as shown in Fig. 2.8. Hop count based objective function has higher control traffic overhead as compared to the ETX and Energy metrics. The control traffic overhead fluctuates in case of PRR, still it can be observed in Fig. 2.9 that ETX has a little bit higher performance than Hop count and Energy.

### 2.4.3 Packet Delivery Ratio

We define the packet delivery ratio as the number of successfully received data packets at the root node divided by the number of data packets transmitted by sensor nodes. The effect of trickle timer parameter is not notable on the packet delivery ratio as shown in the Fig. 2.10. Although both ETX and Energy metrics have better ratio over the Hop count metric. In case of radio duty cycling the performance of Energy metrics is better than both the ETX and Hop count as shown in the Fig. 2.11. However, when the packet reception ratio is varied i.e the packet error rate is increased then the performance of ETX is better than Energy and Hop count metrics as shown in Fig. 2.12. It is due to link quality consideration in ETX. Moreover, as expected the packet delivery ratio increases with the packet reception ratio.

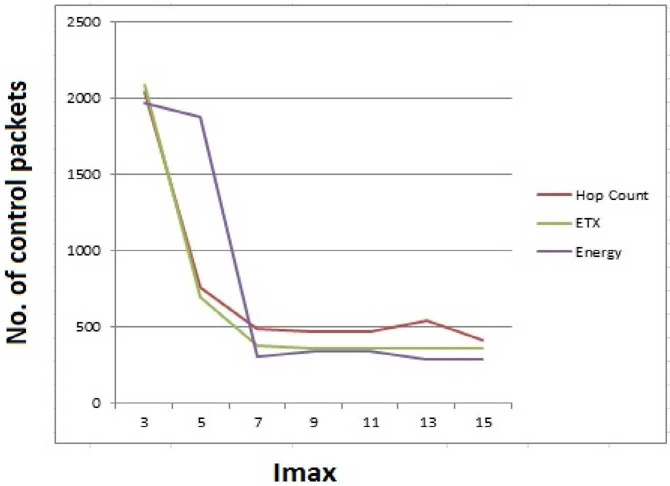


Figure 2.6: DIO MAX vs Control Traffic

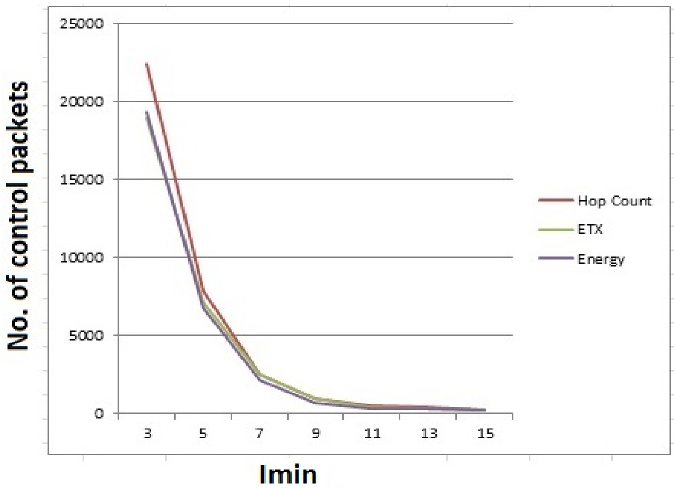


Figure 2.7: DIO MIN vs Control Traffic

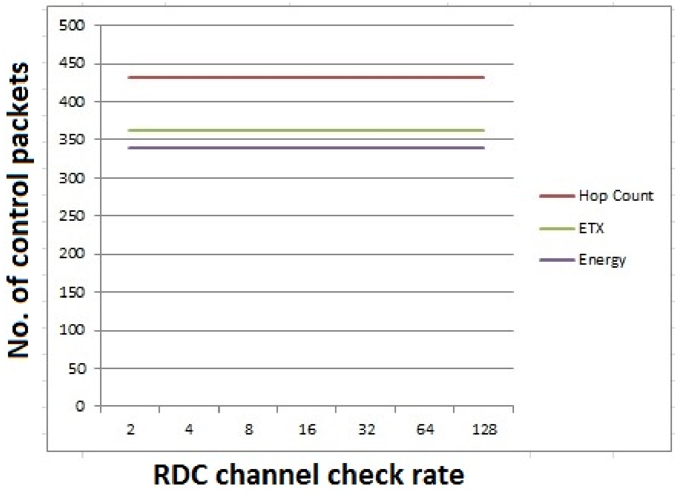


Figure 2.8: RDC vs Control Traffic

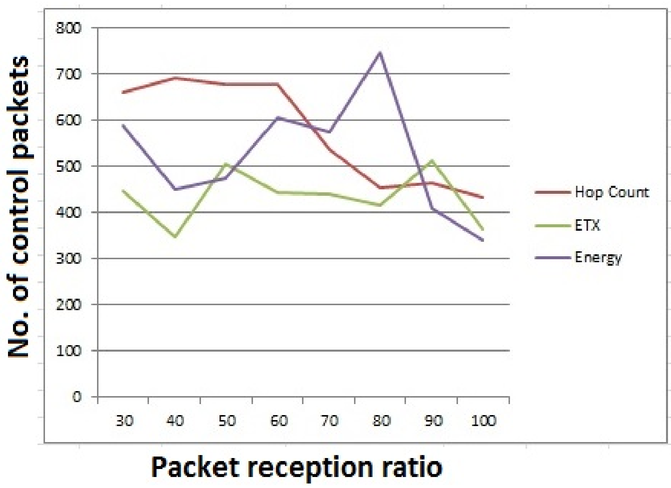


Figure 2.9: PRR vs Control Traffic

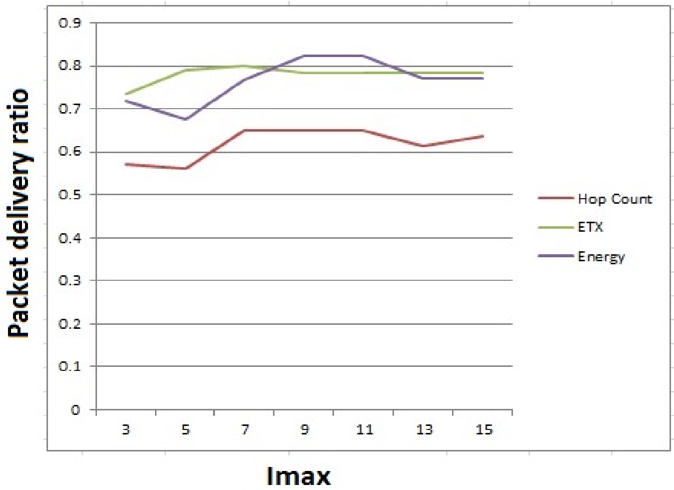


Figure 2.10: DIO MAX vs Packet Delivery Ratio

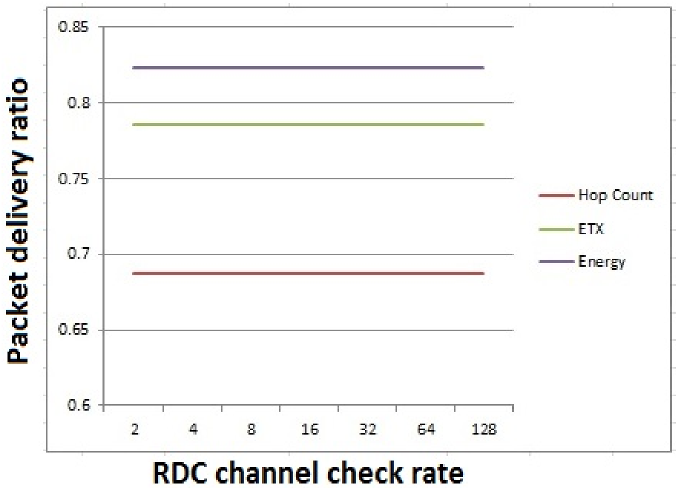


Figure 2.11: RDC vs Packet Delivery Ratio

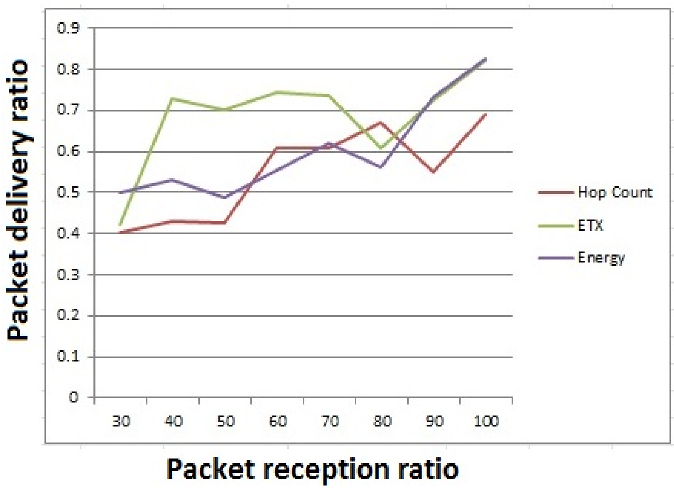


Figure 2.12: PRR vs Packet Delivery Ratio

# Bibliography


[1] Charu C Aggarwal, Naveen Ashish, and Amit Sheth. The internet of things: A survey from the data-centric perspective, managing and mining sensor data. 2013.

[2] Lu Tan and Neng Wang. Future internet: The internet of things. In *Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on*, volume 5, pages V5–376. IEEE, 2010.

[3] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.

[4] Luca Mainetti, Luigi Patrono, and Antonio Vilei. Evolution of wireless sensor networks towards the internet of things: A survey. In *Software, Telecommunications and Computer Networks (SoftCOM), 2011 19th International Conference on*, pages 1–6. IEEE, 2011.

[5] Daniele Miorandi, Sabrina Sicari, Francesco De Pellegrini, and Imrich Chlamtac. Internet of things: Vision, applications and research challenges. *Ad Hoc Networks*, 10(7):1497–1516, 2012.

[6] Ian F Akyildiz, Tommaso Melodia, and Kaushik R Chowdhury. A survey on wireless multimedia sensor networks. *Computer networks*, 51(4):921–960, 2007.

[7] Aura Ganz, Zvi Ganz, and Kitti Wongthavarawat. *Multimedia Wireless Networks: Technologies, Standards and QoS*. Pearson Education, 2003.

[8] P Venkataram and AP Shivaprasad. Wireless multimedia networks. *Journal of the Indian Institute of Science*, 80(3):187, 2013.

[9] Adam Dunkels and Jean-Philippe Vasseur. Ip for smart objects, september 2008. *IPSO Alliance White Paper*, 1.

[10] Vladimir Stankovic, Lina Stankovic, and Samuel Cheng. Compressive video sampling.

[11] Frederic Dufaux, Wen Gao, Stefano Tubaro, and Anthony Vetro. Distributed video coding: trends and perspectives. *Journal on Image and Video Processing*, 2009:10, 2009.

[12] Markus Krüger, Christian U Große, and Pedro José Marrón. Wireless structural health monitoring using mems. *Key Engineering Materials*, 293:625–634, 2005.

[13] Jong-Wan Kim, Hidekuni Takao, Kazuaki Sawada, and Makoto Ishida. Integrated inductors for rf transmitters in cmos/mems smart microsensor systems. *Sensors*, 7(8):1387–1398, 2007.

[14] Routledge. *The Internet of Things: From RFID to the Next-Generation Pervasive Networked Systems*. Routledge, 2008.

[15] G. Montenegro N. Kushalnagar and C. Schumacher. *IPv6 over Low- Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals, Internet Engineering Task Force RFC 4919.* August 2007.

[16] *ZigBee Alliance, Available online: www.zigbee.org.*

[17] IEEE. *802.15.4e-2012: IEEE Standard for Local and Metropolitan Area Networks - Part 15.4: Low-Rate Wireless Personal Area Networks (LRWPANs) Amendment 1: MAC Sublayer*. April 2012.

[18] Jean-Philippe Vasseur and Adam Dunkels. *Interconnecting smart objects with ip: The next internet*. Morgan Kaufmann, 2010.

[19] J. Gettys J. Mogul H. Frystyk L. Masinter P. Leach R. Fielding and T. Berners-Lee. *The Internet of Things: Key Applications and Protocols. HyperText Transfer Protocol - HTTP/1.1, RFC 2616, Internet Engineering Task Force RFC 2616, June 1999. Available online: http://www.rfc-editor.org/rfc/rfc2616.txt.*

[20] J. Postel. *Internet Protocol, RFC 791, Internet Engineering Task Force RFC 791*. September 1981.

[21] *Transmission Control Protocol, RFC 793, Internet Engineering Task Force RFC 793*. September 1981. Available online: http://www.rfc-editor.org/rfc/rfc793.txt.

[22] Philip Levis, T Clausen, J Hui, O Gnawali, and J Ko. The trickle algorithm. *Internet Engineering Task Force, RFC6206*, 2011.

[23] Joydeep Tripathi, Jaudelice Cavalcante de Oliveira, and Jean-Philippe Vasseur. A performance evaluation study of rpl: routing protocol for low power and lossy networks. In *Information Sciences and Systems (CISS), 2010 44th Annual Conference on*, pages 1–6. IEEE, 2010.

[24] N Accettura, LA Grieco, G Boggia, and P Camarda. Performance analysis of the rpl routing protocol. In *Mechatronics (ICM), 2011 IEEE International Conference on*, pages 767–772. IEEE, 2011.

[25] Cheng-Yen Liao, Lin-Huang Chang, Tsung-Han Lee, and Shu-Jan Chen. An energy-efficiency-oriented routing algorithm over rpl.

[26] Olaf Landsiedel, Euhanna Ghadimi, Simon Duquennoy, and Mikael Johansson. Low power, low delay: opportunistic routing meets duty cycling. In *Proceedings of the 11th international conference on Information Processing in Sensor Networks*, pages 185–196. ACM, 2012.

[27] Antimo Barbato, Marica Barrano, Antonio Capone, and Nicolo Figiani. Resource oriented and energy efficient routing protocol for ipv6 wireless sensor networks.

[28] Elnaz Rezaei. Energy efficient rpl routing protocol in smart buildings. 2014.

[29] B. Gronvall A. Dunkels and T. Voigt. Contiki - a lightweight and flexible operating system for tiny networked sensors. In *in Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, pages 455–462. IEEE, 2004.