Performance Analysis of IoT Protocol Stack over Dense and Sparse Mote Network Using COOJA Simulator

Hussam Shamseldin Zenalabdin¹, Abudhahir Buhari¹, Tadiwa Elisha Nyamasvisva¹

¹Infrastructure University Kuala Lumpur, Faculty of Engineering, Science and Technology, Unipark Suria, Jalan Ikram-Uniten, 43000 Kajang, Selangor.Kuala Lumpur, Malaysia.

hussam.shams@yahoo.com

Abstract. Internet of Things (IoT) interrelates computing devices, and digital devices with animals or people and they have ability to transfer amounts of data over a network needless of any human to computer interaction. IoT is composed of sensor nodes or motes, servers, and the internet allowing these devices to collect and exchange information. Lossy networks, stressed out connections and high power consumption are the major concerns in IoT. Protocols are used for messages management and among computing nodes and they act as controls for telecommunication medium using radio frequencies. This paper, evaluates the IoT protocols’ stack over on dense network and sparse network topology which include application layer’s protocols such as, MQTT, CoAP and transport layer protocols’ UDP and TCP. Analysis is done on transport layer through RPL protocol and on physical layer using ZI motes and Sky motes. We utilize Cooja simulator and Contiki operating system for simulations. The performance metrics for the evaluation of protocol stack are power intake, radio responsibility cycle, and average inter-packet arrival time.

1. Introduction
A Wireless Sensor Network (WSN) consists of low small cost devices (nodes) that are characterized by impediments in memory, energy, and capacity. Recently progress in wireless communications and gadgets have enabled the play-out of low-cost, low-power and multi-portability sensors that are small in measurements and communicate in short distances. Frugal and smart sensors are organized and attached wirelessly and spread in wide quantities, feeding unfamiliar opportunities for controlling and monitoring. Networked sensors utilize a wide series of applications inside the protective zone, producing modern capabilities for, observation and different strategic applications. In WSN, power consumption from batteries is restricted, scalability to larger scale of spreading is very high, connectivity flexibility, node mobility, simplicity of use and capability to hold all rough environmental conditions is very high and robust [1] WSN play a vital role in the Internet of thing (IoT).

IoT utilizes ability of connectivity to digitally improved objects usually known as smart things. IoT can be defined as a community of inexpensive, small, low-electricity, encyclopedic electronics gadgets where sensing records and speaking facts manifest without straight human intervention. Each component is able to inter-perform with some of the exiting network infrastructure components and can be uniquely recognizable with the aid of its embedded computing machine. IoT devices are
allowed to cooperate and speak inside themselves with the surroundings. In more than one instances they realize the environment around them and by way of running moves and offerings, responds routinely to physical international incidents. Such smart matters inside the form of services simplify interaction over the network with the assistance of general interfaces. These matters provide a few functionalities that may be known as 'real- world offerings' since few many years the global internet has evolved significantly in numerous ways which has placed sizeable impact on human society as well. Advancement of Internet continues to be on, time to time, with numerous new innovations rising as protocols and so many different inter-wireless verbal exchange technologies may be used to attach the clever gadgets such as MQTT, CoAP, RPL, and many others protocols. Governing operations for IoT devices is the IoT protocol stack. **Error! Reference source not found.** illustrates the IoT protocol stack in which this work is mainly centered on the application layer protocols [2].

![IoT Protocol Stack](image)

**Figure 1. IoT Protocol Stack**

The transport layer protocol, Message Queuing Telemetry Transport (MQTT) is a protocol for publishing / subscribing messaging for lightweight machine to machine communication. Originally developed by IBM, it is an open standard now. MQTT is publishing and subscribe protocol based on TCP developed by IBM, which is then open for messaging applications. In a publish - subscribe
format, customers can either publish data on a specific subject to the server or subscribe to a subject in which the server will automatically send new data to the subscriber after registration. MQTT combines TCP's relatively high overhead and high QoS with the publish or subscribe format capabilities of one-to-one, one-to-many and many-to-one. In addition, this protocol allows customers to specify which telemetry topics are of interest and receive only published data [3].

Another transport layer protocol, CoAP, is utilized as a part of IoT. It has a CoAP architecture, CoAP message header and message trades between CoAP customer and CoAP server. The CoAP protocol is described in RFC 7252. It is a web exchange protocol which is utilized as a part of constrained nodes or networks, for example, WSN, IoT, M2M and so forth. Consequently, the name Constrained Application Protocol. The protocol is aimed for the Internet of Things (IoT) gadgets having limited memory and limited power specifications. As it is intended for web applications it is also called "The Web of Things Protocol". It can be utilized to transport records from a couple of bytes to 1000s of bytes over web applications. It stands between UDP layer and Application layer.

Features of CoAP Protocol: Web protocol from machine to machine restricted; Architecture of representative state transfer (REST); Low overhead for header and complexity of parsing; Multicast supported with reliable unicast and best effort [4-5].

RPL is a distance-vector and a source routing protocol which is intended to work over network layer instruments comprising IEEE 802.15.4 PHY and MAC layers. These join layers can be obliged, conceivably lossy, or in the main used in reference to very obliged host or router border, as an instance, however no longer restrained to, low-electricity wireless or PLC (Power Line Communication) improvements. RPL chiefly objectives accumulation based systems, wherein hubs intermittently ship estimations to an accumulation point. A key aspect of RPL is that it serves as a particular routing resolution for low strength and lossy networks. The protocol was meant to be exceedingly flexible to community environment and to provide trade courses, at something factor default publications are difficult to attain. RPL offers a network to unfold statistics over the powerfuly fashioned network topology. This device utilizes trickle to optimize unfold of manipulate messages.

RPL composes its topology into Direction-Oriented Directed Acyclic Graph (DODAG) or vacation spot Situated coordinated non-cyclic charts. A routing protocol responds to the forwarding of packets from another nodes and smart routing options. Wireless Sensor Network has some sort of routing protocols (Re-active and Pro-active). When desired, the reactive protocols provide routes. It therefore decides whether a path from the mobility of facts is needed by flooding its community question. These varieties of protocols (AODV, DSR and TORA) restrict control traffic and transmit it through the transfer of facts when required. These protocols therefore regularly exchange messages to find and propagate network routes. Nodes send a percentage of local community facts to each local control message; and messages throughout the community to share facts related to the topology among all network nodes. (RPL) is the IPv6 routing protocol for low-power and loss networks designed for low-power and loss networks (ROLL) [5-6].

There is few research have been done so far in performance analysis of MQTT and CoAP protocols in fixed area with various number of motes or sensor nodes. Generally, WSN facing a lot issues, such as power consumption, energy and storage overload while applying in sparse and large scaled network and there is a lack works on RPL routing protocol while using Z1 mote and Sky mote between machine to machine communications. The objectives of the research as follows,

- To compare the performances of MQTT and CoAP (as application layer protocol) over TCP and UDP (as a transport protocol) and RPL (as routing border protocol) in both Sky Mote and Z1 mote (as sensor node) by using the Cooja Simulator.
- To analyse the power consumption, Radio Duty Cycle and Packets Received during network convergence communication.
- To test above configuration on sparse network (15 nodes in 100m x 100m) and dense network (70 nodes in 100m x 100m) topologies.
Table 1. Comparison of Z1 mote and Sky Mote.

|                     | Sky Mote                                                                 | Z1 Mote                                                                 |
|---------------------|--------------------------------------------------------------------------|------------------------------------------------------------------------|
| Sky Mote is the most simple of motes for use within a WSN and ideal for initial configurations within a Cooja simulation. Sky mote is an ultra low power wireless module for use in sensor networks, monitoring applications, and rapid application prototyping [8]. Features Sky mote: -250kbps 2.4GHz IEEE 802.15.4 Chipcon Wireless Transceiver; - Interoperability with other IEEE 802.15.4 devices; -8 MHz Texas Instruments MSP430 microcontrollers (10k RAMS 48 k Flash); - Integrated ADC, DAC, Supply Voltage Supervisor, and DMA Controller. | Z1 mote platform is a wireless sensor network (WSN) development platform for researchers, developers, enthusiasts and amateurs. It is a platform compatible with the successful Tmote family motes with several improvements that offer in several respects approximately 2x performance [9]. Features of Z1 mote: -Ultra-Low Power MCU and 2.4GHz Transceiver; -2 x Digital Built-in sensors (temperature and 3-axis accelerometer); USB Programming Ready; Flexible Powering: Battery Pack (2xAA or 2xAAA), Coin Cell (up to 3.6V), USB Powered. |

2. Literature Review

Table 2. Short literature review:

| Title                                      | Approaches                                      | Limitations                                              | Tools                  |
|--------------------------------------------|-------------------------------------------------|----------------------------------------------------------|------------------------|
| A Comparative Study of MQTT and CoAP Application Layers protocols vis. Performance Evaluation [3]. | Comparing MQTT and CoAP in quality of service levels. | Low data, traffic sensors and high data traffic sensors. | Cooja                  |
| Evaluation Of MQTT and CoAP via common middleware [10]. | Studied the transport of aggregate sensor data from the gateway node to back end server or broker. | The performances of different protocols are different network condition. | Common middleware |
| Performance evaluation of IoT protocols under constrained wireless access network [11]. | In terms of protocols performance study revealing the both TCP-based protocols and MQTT. | Superior performance on data latency and reliability makes it the more attractive candidate. | Wireless Access Network |
| Analysis of energy consumption and topology of routing protocol for low power and lossy network [12]. | Reducing the network energy consumption to the life cycle network on individual network to set more insight on topology. | Balancing node energy consumption and design reasonable routing metrics [13]. | Cooja simulation |

3. Implementation

Cooja is a simulator designed to simulate the operating system of the Contiki sensor network. The simulator is applied in Java, but it allows the program to write sensor node software in C. One of the
differentiating capabilities is that Cooja allows simultaneous simulations at three distinctive levels: Network level, operating system level and training level of machine code. Cooja can also run Contiki applications either natively compiled on the host CPU.

All interactions with the simulated nodes are performed in Cooja via plugins such as Simulation Visualizer, Timeline and Radio Logger. It stores the simulation with an extension 'csc' (Cooja simulation configuration) in an xml report as shown in figure 2 [14].

Figure 2: Cooja Simulator

Cooja is a simulator designed to simulate the operating system of the Contiki sensor network. The simulator is applied in Java, but it allows the program to write sensor node software in C. One of the differentiating capabilities is that Cooja allows simultaneous simulations at three distinctive levels: Network level, operating system level and training level of machine code.

All interactions with the simulated nodes are performed in Cooja via plugins such as Simulation Visualizer, Timeline and Radio Logger. It stores the simulation with an extension 'csc' (Cooja simulation configuration) in an xml report. This record contains information about the simulation environment, plugins, nodes and their positions, random seed and radio medium. The following figure illustrates the flow mechanism of simulation setup. Cooja simulator is one of the active WSN simulator [15-16].
4. Results and Analysis:
This section discusses briefly about the simulation outcome in the perspective of received packets per node, average radio duty cycle and average power consumption. In figure 4. The graph shows number of packets received per node, as seen in Nodes 57 and 59 got highest amount of received 4 packets during transmission because the location far from server, but most of others nodes got 1 packet except nodes 2 and 31 got 2 packets. Figure 5 shows the Average of Radio Duty Cycle. As seen high radio transmit on node 69, got 7.25% Duty Cycle because the cluster of the nodes around it that needs high radio frequency to send and receive. But the remaining nodes got an average between 0.25 to 7.25% for radio listening and radio transmitting. In figure 6 the graph shows the Average Power Consumption, that is the total power consumed after finish testing, Node 69 got 4.90 MW because the highest radio frequency used to cluster of nodes around it. But the remaining got range 0.25 to 4.90 MW power consumption.

![Figure 3. Simulation Flow Chart](image)

![Figure 4. Received Packets Per Node.](image)

![Figure 5. Average Radio Duty Cycle.](image)
Figure 6. Average Power Consumption.

Figure 7. Power Consumption. Figure 8. Radio Duty Cycle.

Figure 9. Received Packets.

Figure 7. shows results in 15 nodes’ case between 1.5 - 1.2 MW for all scenarios, but in 70 nodes’ case shows that Scenario 2 got lowest power consumption 2.50 MW, comparing to Scenario 3 that got the highest power consumption 3.45 MW. Figure 8, shows results in 15 nodes scenarios between 1.4 – 1.2 %, But in 70 nodes scenarios shows that Scenario 2 got the lowest Radio Duty Cycle 3.7%, comparing to Scenario 3 got the highest Radio Duty Cycle 4.25%. But in figure 9. As shows Results in 15 nodes scenarios equally packets received, but in 70 nodes scenarios shows that Scenario 1,2 got highest packets receive 2.6 packets, comparing to Scenario 3,4 that got lowest packets receive 1.82 packets by Throughput = 0.00376/s. (throughput is calculated by dividing the number of packets received over time in seconds)
5. Conclusion
This paper concludes that WSN-based simulation can be done easily with Cooja simulator. Further, Contiki as operating system for every mote mimics the real-world configuration. Moreover, the Contiki offers rich plugins to analyze the results. In this research, total of eight scenarios have been simulated. Among them, scenario 2, which is MQTT -> UDP -> RPL over Sky motes in dense network, outperforms other scenarios. The result correlates with routing in real-world cases. This work can be further enhanced by testing with a large number of motes. As a future work, performance analysis over mobile motes. The mobile motes scenario depicts the detection of fire in forests through the animals which are attached with motes.

References
[1] Tiwari, P., Saxena, V. P., & Mishra, R. G. (2015). Wireless Sensor Networks: Introduction, Advantages. (March 2016).
[2] WIBOONYASAKE, M. (2018). หน้าความรู้จักรใน Internet of Things. Aware Technology Solution for Business.
[3] Oukasse, F. (2018). Comparative Study of Mqtt and Coap Application Layer Protocol Via Performance Evaluation By Faitha Oukasse.Pdf. Journal of Engineering and Applied Sciences. https://doi.org/1816-949
[4] Levä, T., Mazhelis, O., & Suomi, H. (2014). Comparing the cost-efficiency of CoAP and HTTP in Web of Things applications. Decision Support Systems, 63, 23–38. https://doi.org/10.1016/j.dss.2013.09.009
[5] Shelby, Z. (2014). CoAP Standard, 1–112.
[6] Parasuram, A., Culler, D., & Katz, R. (2016). An Analysis of the RPL Routing Standard for Low Power and Lossy Networks. Technical Report No. UCB/EECS-2016-106, 98.
[7] Ali, H. (2012). A Performance Evaluation of RPL in Contiki A Cooja Simulation based study.
[8] Bridge, A. S. N. (2017). Technical University of Crete A Secure Network-Layer Bridge for Wireless Sensor Networks, (June).
[9] Ciabattoni, L., Freddi, A., Longhi, S., Monteriù, A., Pepa, L., & Prist, M. (2016). An Open and Modular Hardware Node for Wireless Sensor and Body Area Networks. Journal of Sensors, 2016. https://doi.org/10.1155/2016/2978073
[10] Thangavel, D., Ma, X., Valera, A., Tan, H. X., & Tan, C. K. Y. (2014). Performance evaluation of MQTT and CoAP via a common middleware. IEEE ISSNIP 2014 - 2014 IEEE 9th International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Conference Proceedings, (April), 21–24. https://doi.org/10.1109/ISSNIP.2014.6827678
[11] Chen, Y., & Kunz, T. (2016). Performance evaluation of IoT protocols under a constrained wireless access network. 2016 International Conference on Selected Topics in Mobile and Wireless Networking, MoWNeT 2016. https://doi.org/10.1109/MoWNet.2016.7496622
[12] Wang, Z. M., Li, W., & Dong, H. L. (2018). Analysis of Energy Consumption and Topology of Routing Protocol for Low-Power and Lossy Networks. Journal of Physics: Conference Series, 1087(5). https://doi.org/10.1088/1742-6596/1087/5/052004
[13] Nyamasvisva, T. E., & Hasbullah, H. (2010). Multi-level security algorithm for random ZigBee Wireless Sensor Networks. Information Technology (ITSim), 2010 International Symposium In, 2, 612–617. https://doi.org/10.1109/ITSIM.2010.5561528
[14] Österlind, F., Dunkels, A., Eriksson, J., Finne, N., & Voigt, T. (2006). Cross-layer sensor network simulation with COOJA. Proceedings - Conference on Local Computer Networks, LCN, 641–648. https://doi.org/10.1109/LCN.2006.322172
[15] Laaouafy, M., Lakrami, F., Labouidya, O., Elkamoun, N., & Iqdour, R. (2019, October). Comparative study of localization methods in WSN using Cooja simulator. In 2019 7th Mediterranean Congress of Telecommunications (CMT) (pp. 1-5). IEEE
[16] Mahmud, A., Hossain, F., Choity, T. A., & Juhin, F. (2020). Simulation and Comparison of RPL, 6LoWPAN, and CoAP Protocols Using Cooja Simulator. Springer, Singapore.