Performance Evaluation of Fuzzy Logic-Based RPL Objective Functions

Alauddin
Department of Computer Science & Information Technology, University of Engineering & Technology, Peshawar, 25000, Pakistan.
***Correspondence:** Alauddin, alauddin_1984@uetpeshawar.edu.pk
Citation | Alauddin, "Performance Evaluation of Fuzzy Logic Based RPL Objective Functions", IJIST, Vol. 6 Issue. 2 pp 918-929, June 2024
Received | June 8, 2024 **Revised |** June 22, 2024 **Accepted |** June 24, 2024 **Published |** June 29, 2024.

Introduction: This paper is based on the evaluation of different fuzzy logic-based approaches, implemented by Routing Protocol for Low-power Lossy networks (RPL), carried out using different topologies.
Importance: This study is carried out to find out the strengths and weaknesses of fuzzy logic-based approaches in RPL for different topologies. Fuzzy logic-based RPL uses a multi-metric approach, i.e., a technique that uses more than one metric for route optimization.
Methodology: Two fuzzy logic-based approaches implemented by RPL are selected, and compared with the single metric techniques, for two different topologies. This comparison is carried out in a network simulator called Cooja. Four performance evaluation metrics, i.e., end-to-end delay, packet delivery ratio (PDR), power consumption, and number of parent switches, are used for comparison.
Novelty Statement: As per the author's knowledge, Evaluation of the fuzzy logic-based RPL techniques for different topologies and the impact of the node's relative location on its results is not carried out.
Results and Discussions: It has been observed that using fuzzy logic in RPL, increases the packet delivery ratio and decreases end-to-end delay and power consumption in some cases. However, at the same time, it increases the number of parents switched. Results also reflected that, in case, there are a small number of nodes i.e., no congestion and the node is closer to the root, instead of using a complicated and time-consuming fuzzy logic-based approach, the originally proposed less-complex methods should be preferred, as they consume less power and also add less processing delay. Fuzzy logic shows better results when the nodes are far away from the root and there is congestion; in this case, a single metric cannot decide the best route for forwarding data.
Concluding Remarks: In future work, while using fuzzy logic in RPL, a dynamic approach may improve the results by selecting an objective function according to the traffic load, number of nodes, and node's location with respect to the root.
Keywords: RPL; Fuzzy Logic; End-to-End Delay; Power Consumption; Objective Functions.

Introduction:

Low-Power and Lossy Network (LLN) is categorized as a network class where LLN routers operate in a constrained environment with limited processing power, battery power, and memory [1]. The IPv6-based Routing Protocol for Low Power and Lossy Networks (RPL), designed by the Internet Engineering Task Force's (IETF) Routing Over Low Power and Lossy Networks (ROLL) group, is tailored for such networks [1]. RPL is commonly used for communication among nodes in IoT (Internet of Things) technology. Before data transmission, rank i.e., node's relative position to a root is calculated. RPL uses objective functions to calculate this rank for each node. RPL is a proactive routing protocol i.e., it starts the creation of routes whenever it is required. Since it is designed for nodes with constrained resources, it is ensured that less memory is utilized during the route's establishment. RPL is categorized as a distance vector protocol [1].

In RPL, a tree-like topology is created, also known as DAG (Directed Acyclic Graph), where all edges point to a single destination, to avoid infinite loops. A node in this DAG, having no outgoing edge is known as a DAG root. This DAG, which has a single DAG root, with no outgoing edge, is termed a Destination-Oriented DAG (DODAG) [1]. Every node selects a parent, based on some predefined parameters, calculated by the sender node and its neighbors. This parent node is then used to forward the message to the destination. The movement of data in RPL is either upward i.e., from leaves to roots, or in a downward direction i.e., from roots to the leaves. The typical RPL topology is shown in Figure 1.

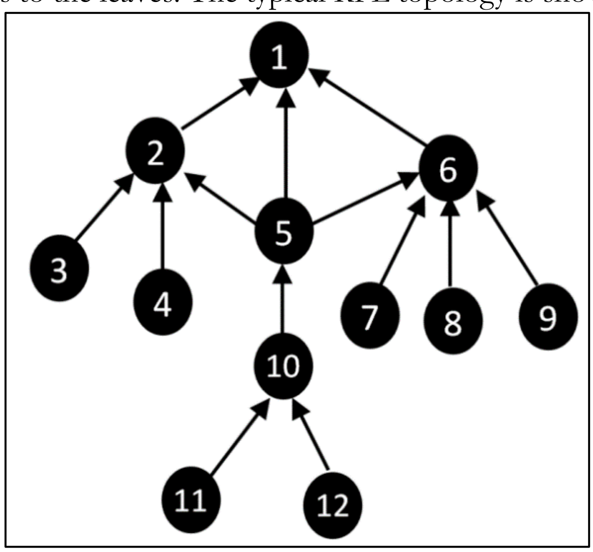


Figure 1. RPL Network topology

In RPL, a root or sink, sometimes also termed an LLN border router (LBR), is used to be connected to an outside network. Some control messages are used by nodes to transfer all the necessary information from the root to other nodes. These control messages are described as DIO (DODAG Information Object), DIS (DODAG Information Solicitation), and DAO (DODAG Destination Advertisement Object). DIO messages are used by the root to discover and form the RPL network. In order to maintain the routing tables, DAO control messages are used. If a node does not receive any DIO message after some specific time, it can send DIS to its neighbor to know if any RPL network is available [1]. In RPL, routing metrics are used by the objective function to find the best route [2]. Traditional networks, like Open Shortest Path First (OSPF), an IP routing protocol use static metrics to find out the best path for directing data traffic. However, RPL gives users the choice of using a variety of dynamic metrics based on the applications and constraints. This feature makes RPL a very attractive protocol for routing in low-power lossy networks. In RPL, metrics are divided into two broader categories, i.e., node metrics and link metrics. Hop count, node energy, etc. are considered as Node metrics while Latency, Link Quality, ETX, are considered as Link Metrics. Contiki RPL implements two routing metrics: hop count and ETX (Expected Transmission Count). [2].

Fuzzy Logic-Based RPL Objective Functions: Related Work:

To enhance the performance of the RPL objective functions, many strategies have been proposed ([3][4][5]). The first strategy uses a single routing metric to select the best route. This single metric can be a node or link metric. Objective function zero (OF0) [6], an objective function that uses a minimum number of hops to find the route toward the destination, is also categorized under this group. Energy total consumed (ENTOT) [7] and energy-based objective function (Energy-OF) [8] are other two examples that use node energy metric to find out the best path. The average delay (AVG-DEL) [9], the received signal strength indicator-based (RSSI-based solution) [10], and the elaborated cross-layer RPL objective function (ELITE) [11] are also some examples that use different link matrices to find out the best route. ELITE [11] measures the number of transmitted strobes per packet in the MAC layer and then reduces the overall transmissions by prioritizing paths that minimize strobe transmissions.

Another effective approach is to combine multiple routing metrics to find the best parent among the available options. Thus, instead of relying on one single metric (i.e., ETX or hop count only), the best route is selected based on the combined information of more than one metric. Fuzzy Logic, a method used to represent and manipulate uncertain and incomplete information, is one of the techniques used for this purpose. Many researchers have applied fuzzy logic techniques to improve the performance of the proposed RPL. Gaddour et al. [12] proposed a modified fuzzy logic-based objective function (OF-FL), combining multiple link and node metrics to provide better QoS. Four metrics; end-to-end delay, ETX, hop count, and node's remaining energy are used in OF-FL. An energy consumption aware objective function (OF-EC), proposed by Lamaazi and Benamaz, also used three metrics: ETX, hop count, and energy consumption, to reduce the network failure probability [4]. In the Fuzzy logic-based energy-aware routing protocol for the Internet of things (FLEA-RPL) [5], Sankar et al. also integrated load, ETX, and residual energy to find the best parent and reduce energy consumption. Araujo et al. [13] used a different approach by introducing four objective functions, Delivery Quality and Context Aware-OFs (DQCA-OFs), which will be selected dynamically based on the information provided by three routing metrics using fuzzy logic. Kamgueu et al., [14] used the expected transmission delay, hop count, and remaining power of a node to get better performance. The results also showed significant improvements. In another research, Aljarrah [15] used nine metrics to select the best parent. To reduce the complexity, different tasks were processed in parallel. In [15] the authors claimed to have better results in terms of end-to-end delay, energy hop count, etc. Kuwelkar and Virani used Residual energy, ETX, and delay to find out the best route [16]. Mehbodniya et al. [17] proposed the Fuzzy Logic Energy Aware Routing Protocol (FLEA-RP), which uses blockchain technology to modify the load on the networks. Results showed that the FLEA-RP protocol improves energy consumption, end-to-end latency, and Network lifetime.[17]

Objectives:

In this research, we have evaluated two fuzzy logic-based objective functions and compared them with the initially implemented Objective Function Zero (OF0), which uses minimum hops for route selections, and Minimum Rank with Hysteresis Objective Function (MRHOF), which uses Expected Transmission Count (ETX) to reach the root. We used a simulator, Cooja, for simulation purposes. Two scenarios have been implemented. Firstly, all nodes were placed linearly in a grid topology, while in the second scenario, all nodes were placed randomly.

Novelty:

Currently, many researchers are incorporating fuzzy logic into RPL routing protocols, highlighting the importance of analyzing the effectiveness of fuzzy logic-based objective functions, especially considering the constraints of IoT devices. As per the author's knowledge, no such research has been carried out that finds the impact of fuzzy logic-based objective functions on RPL for different topologies.

Introduction of Fuzzy Logic: A Soft Computing Method:

Fuzzy Logic is a form of soft computing method that approximates logical reasoning rather than providing an exact solution. In fuzzy logic, variables may have values in a range between 1 and 0, so it does not describe those values as "Yes" or "No", and instead it describes them as "degrees of truthiness". '1' is described as absolutely "True" while '0' is described as absolutely false and a value between 1 and 0 is described as "Degree of Truthiness" [18].

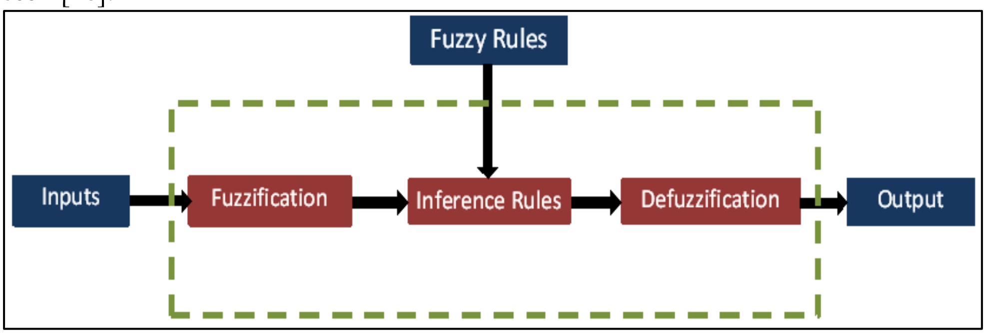


Figure 2. Fuzzy Logic Components

Fuzzy logic components are shown in Figure 2. In order to use a fuzzy logic system, we need input metric data. Some of the most common parameters considered are given below.

Expected Transmission Count (ETX): ETX is an input parameter. It ensures that the path selected by the sender node should have the least number of expected transmissions to reach its destination [2].

Hop-Count: It selects the path with the least hop count to the destination [2].

Power Consumption: This metric is used to calculate the power consumed by the node during transmission. A number of metrics can further be increased; however, it would also increase the number of rules and hence the complexity.

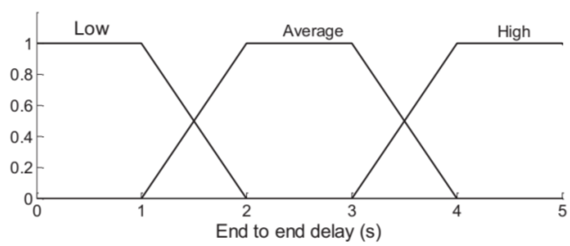
In our evaluation, we have implemented Objective Function Zero (OF0) [6], MRHOF (ETX) [19], OF-FL (fuzzy logic-based RPL) [12], and FAHP-OF [20]. In MRHOF, the expected number of transmissions (ETX) is used as a metric. In OF-FL, four metrics; hop count, ETX, battery level, and end-to-end delay are used for route optimization. In a recent work, Fuzzy Analytic Hierarchy Process Objective Function (FAHP-OF) [20], RSSI is also used along with a scoring mechanism, in which a parent with the highest score is assigned to the child as the next hop for forwarding messages. This score or weight is usually assigned to the input metrics (ETX, RSSI, and Hop-count) by the user, based on network conditions. Thus, in case of congestion the ETX and RSSI may be assigned more weightage to select a safer link for transmission. In [20] ETX is assigned a weight of 0.64, RSSI is assigned 0.07 and Hop count is assigned 0.28 weight. From the results, it can be observed that this assigned weight has very little impact on the nodes that are closer to the root. However, it improves the performance when the distance of nodes becomes larger and also the data traffic increases. [20]

In order to implement the fuzzy logic system, we need some input parameters known as "linguistic variables". These variables are the "words" used to represent a range of information. This information is obtained from the control signals sent by root or neighbor nodes. Then according to the membership functions, they are assigned some degree of membership. Membership functions for end-to-end delay, hop count, ETX, and power consumption are shown in Figure 3(a to d). The explanation for two metrics hop count and ETX is given below in Figure 3(b and c). The distance of a node with respect to the root can be represented using linguistic variables “near”, “vicinity”, and “far”. Figure 3(b) illustrates that nodes having a hop count value below 2 are categorized as “near” to root, and the membership value assigned to these nodes is ‘1’. The number of nodes having values between 2 and 7 are assigned values between 1 and 0. For hop counts 3 and 4, the nodes are assigned a value near 1, and for hop counts 5 and 6, the membership value assigned is close to 0. In other words, we can say that any value of hop count between 2 and 7 defines the degree of closeness to the root [12].

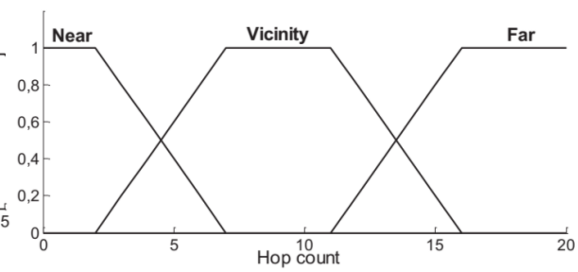
For input variables of ETX, the maximum value 1 (or TRUE) is assigned to ETX values lying between 0 and 10 represented by the linguistic variable “short”. For values between 10 and 30, the “degree of membership” is described as “short”. The linguistic variable “average” is represented by ETX values ranging from 20 to 80, with the maximum degree of membership (1) assigned to ETX values between 30 and 60. For the remaining values, the degree is calculated. For the variable “average”, any ETX value above 80 is described as 0 (false). Any ETX value greater than 80 has a 1 (true) value for the variable “large”. The range of ETX values and their mapping to the linguistic variables are totally based on the simulation results mentioned in [12].

Membership Functions:

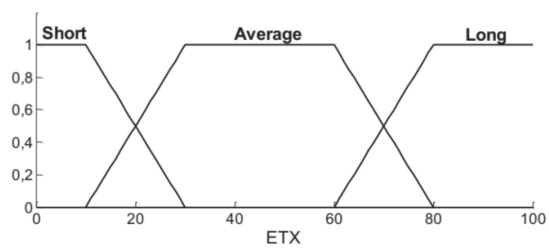
To quantify the linguistic variables, we use membership functions. Trapezoidal graphs are commonly used for this purpose in fuzzy logic. The definition of membership functions for the end-to-end delay, hop count, ETX, and battery level implemented by [12] are shown in Figure 3.



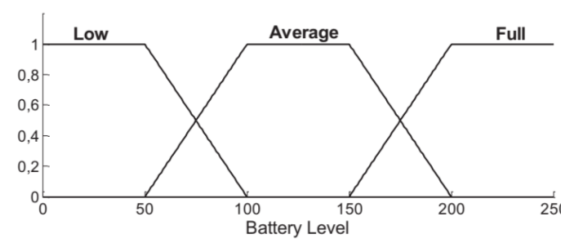
a) Input variable for End-to-end delay.



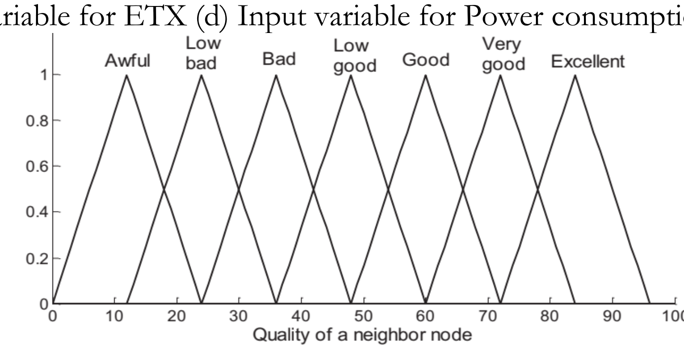
b) Input variable for the hop count [12]



c) Input variable for ETX



(d) Input variable for Power consumption [12]



(e)Output variable of a neighbor quality

Figure 3. Definition of membership functions [12]

Equations of Degree of Membership (DOM) for one of the metrics ETX are given below [12];

$$small(path_{etx}) = \begin{cases} 1 & \text{if } path_{etx} \leq 10 \\ \dfrac{path_{etx} - 30}{10 - 30} & \text{if } 10 < path_{etx} < 30 \\ 0 & \text{if } path_{etx} \geq 30 \end{cases}$$

$$average(path_{etx}) = \begin{cases} 0 & \text{if } path_{etx} \leq 10 \\ \dfrac{path_{etx} - 10}{30 - 10} & \text{if } 10 < path_{etx} < 30 \\ 1 & \text{if } 30 \leq path_{etx} \leq 60 \\ \dfrac{path_{etx} - 80}{60 - 80} & \text{if } 60 \leq path_{etx} \leq 80 \\ 0 & \text{if } path_{etx} \geq 80 \end{cases}$$

$$large(path\_etx) = \begin{cases} 0 & \text{if } path\_etx \leq 80 \\ \dfrac{path\_etx - 60}{80 - 60} & \text{if } 60 < path\_etx < 80 \\ 1 & \text{if } path\_etx \geq 0 \end{cases}$$

Like ETX, DOM for all the metrics used by any fuzzy logic must be defined as mentioned in [12] and [20]. After defining DOM functions, the next step involves deriving rules to give meaning to all combinations of linguistic input variables. In OF-FL [12] four input metrics with 4 membership functions are implemented. Thus, the total number of rules is 4^4=256. Some of them are shown in the table 1. [12]

Table 1: Fuzzy rule base ([12],[21])

Hop Count	End-to-end delay	Battery level	ETX	Quality
Near	low	high	short	Excellent
Near	low	high	average	very good
Near	low	average	short	very good
Near	average	high	long	good
Vicinity	low	average	average	good
Vicinity	average	high	short	low good
Vicinity	high	low	average	bad
Vicinity	average	average	long	bad
Far	high	high	average	low bad
Far	high	average	short	low bad
Far	average	low	average	awful
Far	high	low	long	awful

In Table 1, the neighbor quality, i.e., the output fuzzy variable is shown in the last column. This output fuzzy variable is composed of five fuzzy sets, from “awful” to “excellent." A numerical value between 0 and 100 is used to represent these output variables and is assigned accordingly to a neighbor node [20] as shown in Figure 3(e).

The neighbor with the best quality will be selected as a parent. The neighbor with the best energy level, a smaller number of hops counts to the root, and a smaller value of ETX will be considered the node with the best quality. For the evaluation of these rules and finding their output, we use Mamdani implications [18].

Defuzzification:

In this step, we find a crisp value for a range of fuzzy sets. The centroid method [21] is implemented in [20]. In this method, a balance point is calculated by the weighted mean of the fuzzy region.

Methodology:

In our evaluation process, we have simulated objective functions OF0, MRHOF, OF-FL [12], and FAHP-OF [20]. The main reasons for the selection of these four specific objective functions for comparison and evaluation are given as under. OF0 and MRHOF are the two objective functions, proposed and implemented initially by the Contiki OS. Therefore, any new proposed version of RPL is usually compared with OF0 and MRHOF. OF-FL is considered for evaluation as it is one of the earlier proposed fuzzy logic-based objective functions [12] and hence can be described as a state-of-the-art objective function for fuzzy logic. FAHP-OF [20], is another proposed approach that introduced the weighted version of the fuzzy logic-based mechanism.

We have carried out these simulations using a popular Contiki OS-based network simulator, Cooja. Cooja is used for the simulation of IoT networks [22]. The Cooja simulation parameters are shown in Table 2. An area of 180 x 180 meters is selected for arranging nodes. The selection of area and number of nodes is scenario-dependent; it can be modified according to the requirements. The simulation run time is 1 Hour. Again, it can be changed, providing enough simulation time to get all the necessary results. Nodes are placed at (i) random and (ii) linear or grid positions, as shown in Figure 4.

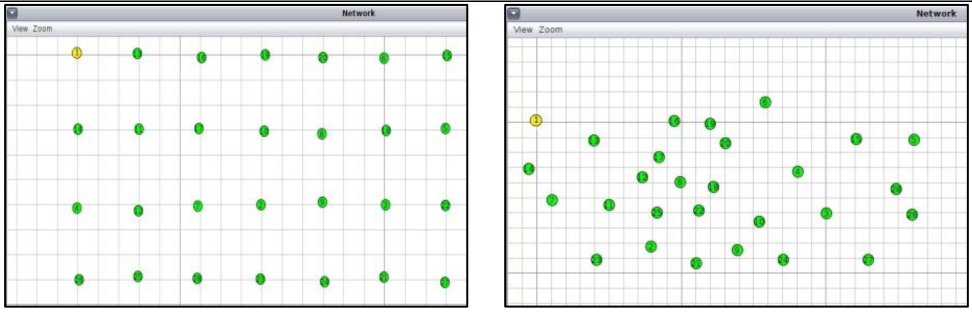
Table 2: Simulation Parameters

Parameters	Values
OS	Contiki-ng
Model	UDGM
Start Delay	65 s
No. of Nodes	28
Area	180 x 180 m sq
Objective Functions	MRHOF(ETX), OF0, FL-OF, FAHP-OF
Simulation Time	1 Hr (3600 Seconds)

The following metrics are used for the analysis and evaluation of objective functions:

- End-to-end delay: It is the time taken by a packet from the sender to the root. In our simulations, the delay for the different objective functions is calculated, and the results are discussed.
- Packet delivery ratio (PDR): It is the ratio of the total number of delivered packets (at the DAG root) to the total number of sent packets (by all the router nodes).
- Average power consumption: This metric shows how much average power is consumed by each node during transmissions. It can also be used to estimate the network lifetime.
- Average number of parent changes: It shows the total number of times a node has switched its parents during the transmission of messages to the root.

The main reason for selecting the above-mentioned four metrics is that they can give us better information regarding the network performance. For example, end-to-end delay is an important metric to find out the delay a node can face while sending a packet from sender to destination. PDR is used to find out the reliability of the network. A higher value of PDR means the network is more reliable. Average power consumption is also very important, as RPL is implemented mostly by IoT devices so calculating the power consumption for each objective function and then their evaluation is very important. Calculation of Parent switching during data transmission is also very useful. A larger value of this metric describes the instability of a network. A network will be considered more reliable and stable if the node does not change its parent frequently.



(a) Nodes in linear/grid topology (b) Nodes in random topology

Figure 4. Topologies considered in the simulation.

The methodology for the evaluation of fuzzy logic-based OFs is shown in Figure 5. In this flow diagram, it is shown that after placing the nodes the root initiates the creation of the RPL network. After the creation of the RPL network each node applies fuzzy logic to select its preferred parent. This implementation is carried out by the extraction of metric parameters (power consumed, ETX, and Hop count) from the control message. After extraction, the processes of fuzzification and defuzzification are carried out. Then the preferred parent is selected and the data is sent to the root via the new preferred parent.

Placement of nodes and roots according to the topologies mentioned in fig 4 (a & b)
↓
Initialization of Root for the creation of RPL Network
↓
Implementation of Fuzzy logic for the selection of preferred parent by each node
↓
Extraction of inputs (power consumed, ETX, hop count) from the control messages
↓
Fuzzification
↓
Application of Fuzzy rules
↓
Defuzzification
↓
Calculation of Output showing the preferred parent quality
↓
Sending the data to Root via new selected parent
↓
Comparison with OF0 and MRHOF

Figure 5. Flow diagram for the methodology of Evaluation of fuzzy logic-based OFs

Results:

The simulation setup is shown in Fig 4. Nodes with yellow color are the root nodes. In (a), nodes are arranged in grid topology with the root at the leftmost corner (Root No. 1). In (b), nodes are randomly deployed; again, the root is located at the left corner instead of the center. The placement of the root at these positions is merely to see the performance of these algorithms in worse situations. Each DAG root initiates the creation of a DODAG network and makes an attempt to add all the reachable nodes. On the other hand, each node also searches for the availability of the network by sending control messages to its neighbors. Upon receiving a message from any connected node, a disconnected node registers itself with that root. If a node receives multiple DIOs from more than one node connected with different roots, it calculates its own rank and then, using fuzzy logic finds the best route to the root.

Average End-to-End Delay:

It is clear from Figures 6(a) and 6(b) that when nodes are placed in a grid topology, OF0 exhibits maximum latency as compared to other nodes. At a distance of 0 to 60 meters, this difference in latency is not noticeable; however, as the distance of the nodes increases from the root, this end-to-end delay also increases. On the other hand, at even larger distances, the delay produced by fuzzy logic-based objective functions becomes smaller as compared to OF0 and MRHOF. In the case of randomly placed nodes, the pattern of delay is almost the same; for nodes near the root, the delay is almost the same for all the objective functions. However, as the distance from the root increases, the fuzzy logic-based OFs perform better than OF0 and MRHOF. The poor performance of OF0 can be attributed to its disregard for link quality during data transmission, leading to more transmissions and, consequently, greater delay. Delays introduced by fuzzy logic-based Objective functions are also understandable, as it takes more processing time to calculate the best parent; hence, the best route is selected at the cost of some extra delay.

Packet Delivery Ratio (PDR):

It can be seen from simulation results (Figures 7 (a) and (b)) that fuzzy logic-based OFs show the best PDR, both for grid topology and random topology. This is because fuzzy algorithms consider link quality along with node quality. On the other hand, OF0 has the smallest PDR as it does not consider the link quality. Results also show that nodes near the root have the highest PDR for all techniques evaluated; however, as the distance from the root increases, only fuzzy logic-based algorithms perform better.

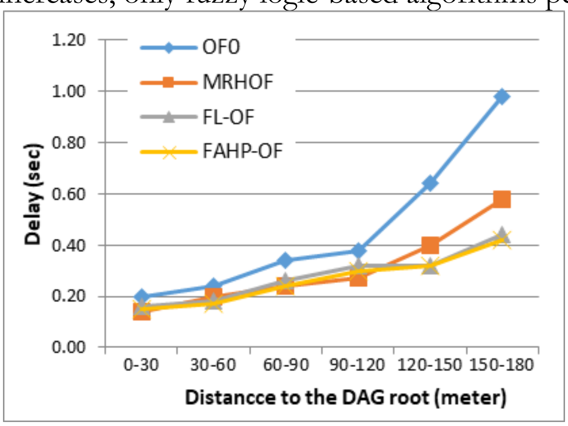


Figure 6 (a). Delay in grid topology

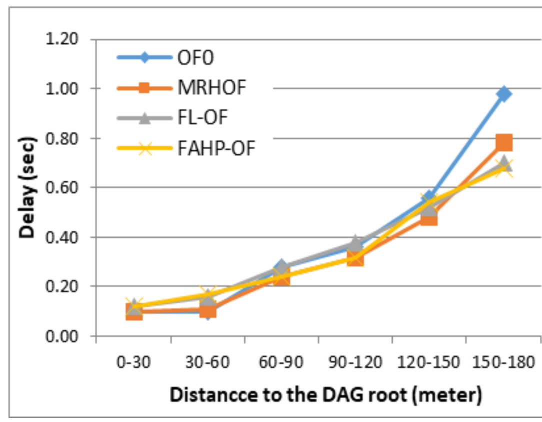


Figure 6 (b). Delay in Random topology

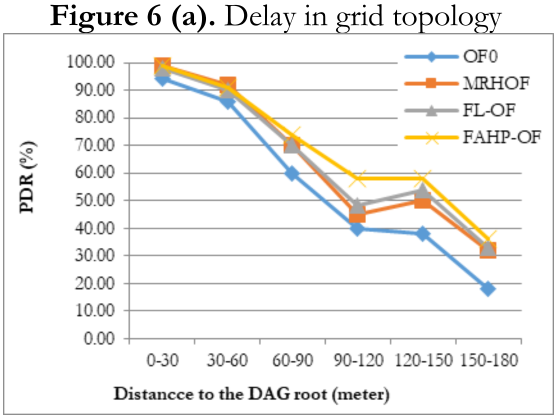


Figure 7 (a). PDR in a grid topology

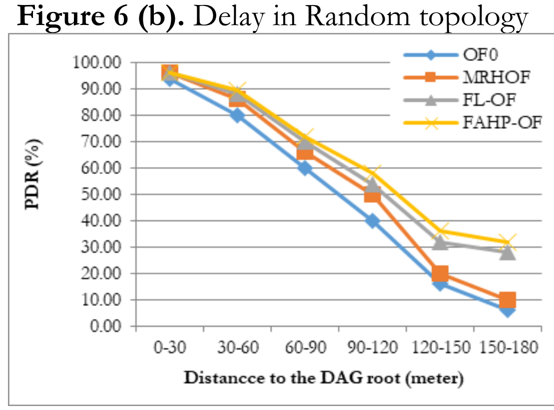


Figure 7 (b). PDR in Random topology

Power Consumption:

Considering the limited resources and constraints of IoT devices, power consumption is a very important metric for evaluating the performance of any algorithm. From our simulation results (Figures 8 (a) and (b)), it can be concluded that nodes closer to the root consume almost the same power, whatever objective function is used. However, fuzzy logic-based RPL protocols consume much less power when the distance from the root increases. This reduction in power consumption by fuzzy logic-based OFs is because it reduces unnecessary transmission of packets.

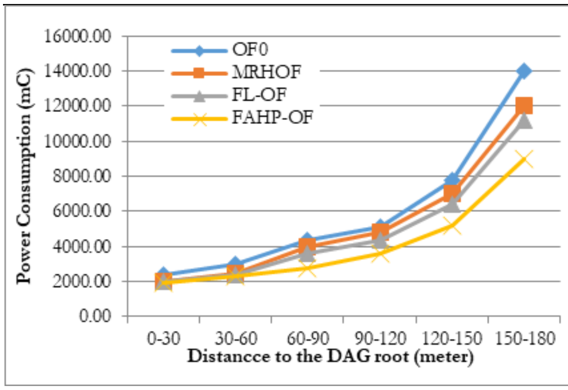


Figure 8 (a). Power Consumption in Grid Topology

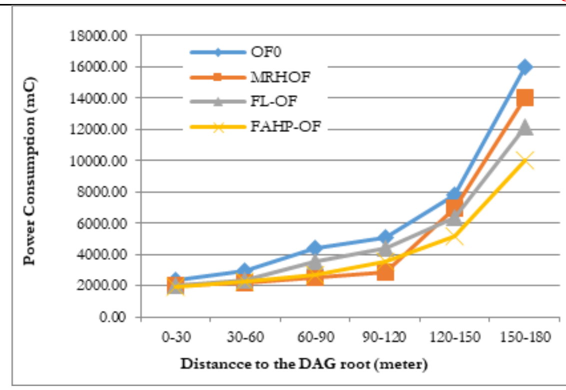


Figure 8 (b). Power Consumption in Random Topology

Parent Switching:

The most interesting results are shown by the parent switching graphs in Figure 9 (a & b). A high frequency of parent node switching is undesirable as it leads to network instability. One drawback of fuzzy logic is that a relatively larger number of parents switching is observed. However, this is acceptable within a specific threshold to achieve higher PDR, smaller delay, and less power consumption. It is also interesting to observe that parent switching is relatively smaller when the nodes are closer or very far from the root. Nodes in the range of 30 to 120 meters show high traffic and therefore result in more switching. The smaller values of parent switching reported by the nodes closer to the root are due to the fact that these nodes have very few options for selecting their parent, as the only option they have is the root itself, which reduces the switching process.

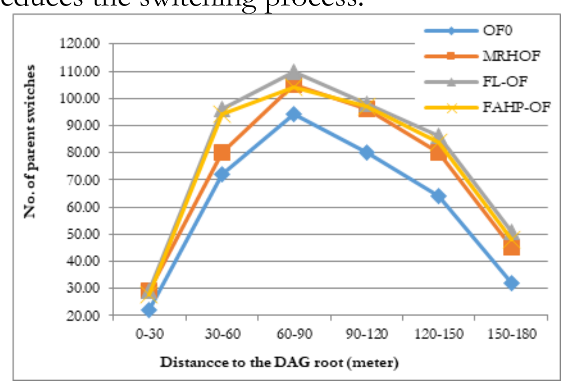


Figure 9 (a). Parent switching in grid topology

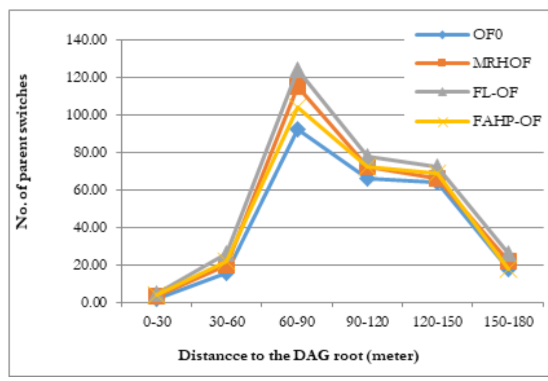


Figure 9 (b). Parent switching in Random topology

Discussion:

From the results, it is clear that the end-to-end delay produced by the nodes is almost the same for all the OFs, provided, the sender nodes are close to the root. This means that for nodes located near the root, we can implement any objective function and by this, their performance will not be affected to a large extent. However, with the increase in distance from the root, the performance of the OF0 and MRHOF becomes unsatisfactory in terms of delay and PDR, while the performance of fuzzy logic-based OFs improves significantly.

Power consumption and parent switching attributes are also affected by the number of neighbors and distance concerning the root. It is clear from the results that all objective functions show almost the same performance when nodes are closer to the root in terms of parent switching and power consumption; however, with the increase in distance and number of neighbors role of OFs becomes more crucial. Performance of OF0 and MRHOF become very poor in areas where the number of nodes increases and also the distance between the sender node and receiver increases. Parent switching is slightly undesirable in a fuzzy logic-based approach, but it is a fact that this complexity may be accepted to achieve better PDR, smaller delay, and less power consumption. The better results shown by FAHP-OF are due to the assigned weights to input metrics. These weights are assigned according to the calculations mentioned in [20].

It can be seen that by assigning more weightage to the ETX, the PDR, and other parameters showed better results. All these results are summarized in Table 3

Table 3: Values of PDR, Power consumed, Latency, and Parents switched for OF0, MRHOF, OF-FL, and FAHP OFs when nodes are placed at different locations w.r.t root.

	OF0			MRHOF			OF-FL			FAHP-OF		
Distance w.r.t Root (meter)	0-60	60-120	120-180	0-60	60-120	120-180	0-60	60-120	120-180	0-60	60-120	120-180
PDR (%)	100-75	75-45	45-20	100-80	80-50	50-30	100-80	80-50	50-35	100-85	85-60	60-35
Power ($x10^3$ mc)	2.4-3.0	3.0-5.1	5.1-14	2.0-2.5	2.5-4.8	4.8-12	2.0-2.4	2.4-4.4	4.4-11	1.9-2.3	2.3-3.6	3.6-9
Latency (Sec)	.20-.24	.24-.38	.38-.98	.14-.20	.20-.27	.27-.58	.16-.18	.18-.32	.32-.44	.15-.17	.17-.30	.30-.42
Parents Switched	22-72	72-80	80-32	29-80	80-96	96-45	29-96	96-98	98-51	28-94	94-97	97-48

Conclusion:

It is clear from the simulation results that fuzzy logic-based RPL shows good results in high-traffic scenarios and when the senders are far away from the root. All those nodes that are nearer to the root exhibit almost the same results for all the OFs. Therefore, a dynamic approach needs to be adopted so that nodes near the root may avoid complex fuzzy logic-based calculations and use simple OF0 or MRHOF for selecting the next hop. While nodes far from the root or located in a congested area should select fuzzy logic-based OFs to find the best route to the root. Another tradeoff reflected by these results is that FL-OF and FAHP-OF introduce some delay, which is not desirable in situations where real-time data is required. However, sending data without considering other link metrics can also cause unnecessary re-transmissions or loss of data. Therefore, the use of fuzzy logic is still preferred to ensure successful delivery to the root at the cost of some acceptable delay.

Acknowledgment: I am thankful for the guidance and support of the administration and lab staff of the Secured IoT Devices Lab, Department of Computer Systems Engineering, University of Engineering and Technology, Peshawar, regarding my research.

Funding: No external funding is received.

Conflict of Interest: The author declares no conflict of interest.

References:

[1] A. Brandt et al., “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” Mar. 2012, doi: 10.17487/RFC6550.

[2] K. Pister, N. Dejean, and D. Barthel, “Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks,” Mar. 2012, doi: 10.17487/RFC6551.

[3] “FSS: Fuzzy supervised learning for optimal path selection in RPL”, [Online]. Available: https://iopscience.iop.org/article/10.1088/1742-6596/1911/1/012016

[4] H. Lamaazi and N. Benamar, “OF-EC: A novel energy consumption aware objective function for RPL based on fuzzy logic.,” J. Netw. Comput. Appl., vol. 117, pp. 42–58, Sep. 2018, doi: 10.1016/J.JNCA.2018.05.015.

[5] S. Sankar and P. Srinivasan, "Fuzzy logic based energy aware routing protocol for internet of things," Int. J. Intell. Syst. Appl., vol. 10, no. 10, p. 11, 2018, doi: 10.5815/IJISA.2018.10.02.
[6] P. Thubert, Ed., "Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)," Mar. 2012, doi: 10.17487/RFC6552.
[7] F. Demicheli, "Design, Implementation and Evaluation of an Energy Efficient RPL Routing Metric," 2011, [Online]. Available: https://www.tesionline.it/tesi/thesis-author.jsp/45377?idt=45377
[8] P. O. Kamgueu, E. Nataf, and T. N. Djotio, "On design and deployment of fuzzy-based metric for routing in low-power and lossy networks," Proc. - Conf. Local Comput. Networks, LCN, vol. 2015-December, pp. 789–795, Dec. 2015, doi: 10.1109/LCNW.2015.7365929.
[9] G. Gonizzi, P.; Monica, R.; Ferrari, "Design and evaluation of a delay-efficient RPL routing metric," pp. 1573–1577, [Online]. Available: https://ieeexplore.ieee.org/document/6583790
[10] Tsung-Han Lee; Xiang-Shen Xie; Lin-Huang Chang, "RSSI-based IPv6 routing metrics for RPL in low-power and Lossy networks", [Online]. Available: https://ieeexplore.ieee.org/document/6974164
[11] Bardia Safaei; Amir Mahdi Hosseini Monazzah; Alireza Ejlali, "ELITE: An Elaborated Cross-Layer RPL Objective Function to Achieve Energy Efficiency in Internet-of-Things Devices," pp. 1169–1182, [Online]. Available: https://ieeexplore.ieee.org/document/9148607/authors#authors
[12] O. Gaddour, A. Koubǎa, N. Baccour, and M. Abid, "OF-FL: QoS-aware fuzzy-logic objective function for the RPL routing protocol," 2014 12th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wirel. Networks, WiOpt 2014, pp. 365–372, 2014, doi: 10.1109/WIOPT.2014.6850321.
[13] H. da S. Araújo et al., "A Proposal for IoT Dynamic Routes Selection Based on Contextual Information," Sensors 2018, Vol. 18, Page 353, vol. 18, no. 2, p. 353, Jan. 2018, doi: 10.3390/S18020353.
[14] "Fuzzy-based routing metrics combination for RPL", [Online]. Available: https://inria.hal.science/hal-01093965/document
[15] E. Aljarrah, "Deployment of Multi-Fuzzy Model Based Routing in RPL to Support Efficient IoT," Int. J. Commun. Networks Inf. Secur., vol. 9, no. 3, pp. 457–465, Dec. 2017, doi: 10.17762/IJCNIS.V9I3.2918.
[16] S. Kuwelkar and H. G. Virani, "Design of an Efficient RPL Objective Function for Internet of Things Applications," IJACSA) Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 6, p. 2021, Accessed: Jun. 11, 2024. [Online]. Available: www.ijacsa.thesai.org
[17] A. Mehbodniya et al., "Energy-Aware Routing Protocol with Fuzzy Logic in Industrial Internet of Things with Blockchain Technology," Wirel. Commun. Mob. Comput., vol. 2022, no. 1, p. 7665931, Jan. 2022, doi: 10.1155/2022/7665931.
[18] "A Study of Membership Functions on Mamdani Type Fuzzy Inference System for Industrial Decision-Making", [Online]. Available: https://core.ac.uk/download/pdf/228639047.pdf
[19] O. Gnawali and P. Levis, "The Minimum Rank with Hysteresis Objective Function," Sep. 2012, doi: 10.17487/RFC6719.
[20] M. Koosha, B. Farzaneh, E. Alizadeh, and S. Farzaneh, "FAHP-OF: A New Method for Load Balancing in RPL-based Internet of Things (IoT)," 2022 12th Int. Conf. Comput. Knowl. Eng. ICCKE 2022, pp. 471–476, 2022, doi: 10.1109/ICCKE57176.2022.9960073.
[21] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," IEEE Trans. Syst. Man Cybern., vol. SMC-15, no. 1, pp. 116–132, 1985, doi: 10.1109/TSMC.1985.6313399.
[22] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki - A lightweight and flexible operating system for tiny networked sensors," Proc. - Conf. Local Comput. Networks, LCN, pp. 455–462, 2004, doi: 10.1109/LCN.2004.38.