

A NOVEL STABLE PATH SELECTION ALGORITHM FOR ENHANCING QoS AND NETWORK LIFETIME IN RPL-CONTIKI- BASED IOT NETWORKS

Mohamed Achref BOUKHOBZA ¹, Mehdi ROUISSAT ^{1,2}, Mohammed
BELKHEIR ³, Allel MOKADDEM ³ and Pascal LORENZ ⁴

¹ NourBachir University Center, El-Bayadh, Algeria

² STIC Laboratory, University Aboubekr Belkaid, Tlemcen 13000, Algeria

³ LIMA Laboratory, Nour Bachir University Center El-Bayadh, Algeria

⁴ Haute Alsace University, Mulhouse, France

ABSTRACT

The Internet of Things (IoT) facilitates real-time connectivity of objects, allowing for access from anywhere at any time. For IoT Low-Power and Lossy Networks (LLNs), the Routing Protocol for Low-Power and Lossy Networks (RPL) has been introduced. In RPL-based topologies, the rank of nodes reflects their positions within the network, calculated by adding the rank of a node's preferred parent to the link metric between them. However, due to inaccuracies in assigning link metric values to neighboring nodes, frequent changes in preferred parent selection occur, resulting in significant control overhead, increased energy consumption, higher latency, and degraded Packet Delivery Ratio (PDR). This paper presents an optimized path selection method that ensures the most stable and optimal choice of preferred parents for nodes. Using the Cooja simulator under various network densities, the proposed approach demonstrates a 73% reduction in preferred parent changes, a 49% decrease in control overhead, and a 50% reduction in total energy consumption. Additionally, it improves PDR by 46% and reduces latency by 2.81 seconds.

KEYWORDS

IoT, RPL, QoS, Objective Functions, Routing metric, ETX

1. INTRODUCTION

IoT technologies have become ubiquitous in all facets of our everyday life, driving a revolutionary technological change in our society in recent years. An enormous change has occurred with the introduction of IoT networks, which have produced self-automated settings designed to make process data interchange easy. This has been crucial in making use of internet capabilities. Wireless Sensor Networks (WSNs) play a crucial role in IoT by handling data collection from various smart emerging applications such as smart cities [1] to smart healthcare systems [2], smart vehicular networks [3-4], smart farms and smart agriculture practices [5-6], and various other smart industrial applications [6-8], have revolutionized efficiency and services by enabling real-time data collection and transmission. This technological evolution owes much to the emergence of small-sized smart devices, known as smart sensors, designed to monitor specific environments and gather pertinent data [9]. Simultaneously, cloud computing platforms has provided substantial processing capacity as well as a reliable storing place, enabling the storage of vast data volumes for subsequent processing and decision-making [10-11].

IoT applications, comprising multiple components, present substantial challenges in terms of resource optimization, given the constraints of energy, routing, and memory. The ROLL working group has addressed the demand for routing protocols in IoT development by creating RPL, a commonly used routing protocol. RPL employs two objective functions, which are not algorithms but rather optimized paths based on metrics. OF0 employs hop count as the metric, while MRHOF uses ETX as a core metric.

Over the past decade, research and IoT application development in WSNs have increasingly focused on IPv6 routing for low energy consumption and minimized packet loss rates. This effort aligns with the specific requirements of low energy consumption and Quality of Service (QoS) in IPv6 routing for Low-power Wireless Personal Area Network (6LowPAN) [12]. This approach integrates the IEEE 802.15.4 standard and is known as the Routing Protocol for Low-Power and Lossy Networks (RPL). RPL serves as the standard routing protocol for IPv6-based Low-Power and Lossy Networks (LLNs), such as WSNs. It is presented by the ROLL working group and operates at the IEEE 802.15.4 Physical and Data Link layers in a dynamic and extensible routing manner, computing path distances using distance-vector routing. In contemporary LLNs, there is a pressing need for rapid access to diverse information from the Internet and sensor devices, connecting the physical world with networked information. However, several areas of improvement remain the focus of ongoing research. Due to the inherent constraints of components in this network, routing in LLNs is a challenging task. Standardization groups and researchers worldwide are dedicated to designing high-quality routing protocols for IPv6-based LLNs. RPL, while flexible, employs a routing metric that may not fully address link reliability. [13] Compared to other networks, IoT devices are intelligent but still require substantial enhancements in routing protocols. Specifically, in the context of the RPL protocol, optimizing constrained resources is a significant challenge. This optimization encompasses route formation, energy efficiency, enhanced transmission speed, node interconnection, and addressing bottleneck issues. The need to address lossy and unstable links that result in lower data rates in WSNs underscores the potential for advancing the RPL protocol to the next level. There is considerable scope for optimizing constrained resources within RPL [14].

The default OF in RPL Contiki is called MRHOF, which determines a node's rank mostly on the quality of its path from a given node to the sink node. The neighbor with the lowest route metric is chosen to be the preferred parent, and the choice of preferred parent is based on which neighbor node has the best path metric. The rank value that a neighbor node advertises and the link metric value that is connected to it determine the path metric value of that neighbor node. The best path metric value, as determined by calculating it through all of the candidates, is used to determine which parent is selected. The possible rank value that the child node may receive if the candidate parent is chosen is given by the route metric across a particular node. As a result, a key factor in determining the rank is the rank growth, or link metric. RPL-Contiki uses 512 as the starting value for each new neighbor, this particular static value engenders an instability and continuous selection of the preferred parents, that leads to a significant exchange of control overhead, consumed energy, latency as well as a degradation in the PDR. To the best of our knowledge, this is the first paper that study details the parent switching process in RPL-Contiki, and highlight the limits of RPL-Contiki implementation in the point. Thus, the aim of this paper is giving a detailed analysis on the principle of parent witching in RPL-Contiki and define a verified, dynamic and precise value to the link metric that reflect the position of the nodes within the topology.

2. RPL PROTOCOL AND ETX OVERVIEW

1.1. RPL Protocol

IETF has officially designated RPL as the primary routing protocol for Low-power and Lossy networks, in RFC 6550. RPL operates on the foundation of the IPV6 protocol and adopts a mesh/tree topology, culminating in the formation of an acyclic graph known as a DODAG (Destination Oriented Directed Acyclic Graph). This DODAG structure is originated from a root node, symbolizing the data sink, and extends to other nodes based on their assigned rankings within the network. To oversee the fundamental routing processes and manage the DODAG, RPL introduces four crucial control messages: DIS, DIO, DAO, and DAO-ACK messages [15], as illustrated in Figure 1.

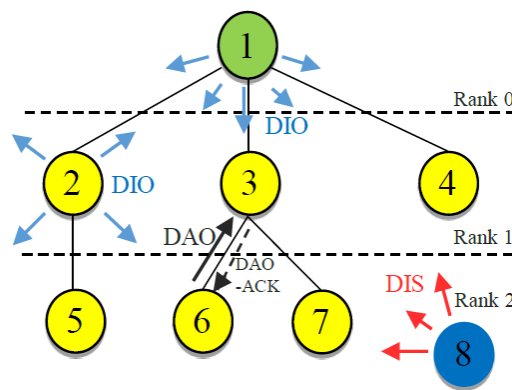


Figure1. RPL Topology and Control Messages

To join the network, every node must initiate a request for a set of repositories from neighboring nodes. The request is made by sending a DIS (DODAG Information Solicitation) message. In response to a received DIS message, nodes send a DIO (DODAG Information Object) message. The DIO message includes essential topology parameters such as the DODAG ID, Version Number, Rank value, and MOP (Mode of Operation). These parameters are under the jurisdiction of the root node and are pivotal for enabling each node to become part of the DODAG. DIO messages play a key role in constructing upward routes from nodes to the root. For building downward routes, RPL employs DAO messages (Destination Advertised Object) to disseminate additional information, including parent-child relationships and other necessary parameters required for propagating destination information downward the DODAG [16]. During regular operation, the root node initiates a trickle timer, which is then propagated throughout the network to regulate the periodic exchange of DIO messages, aiming to optimize the exchange of control overhead. RPL incorporates a repair mechanism known as "local repair," which is initiated by individual nodes to update routes or in response to node-related issues. Additionally, the root node could initiate a global repair mechanism by incrementing the current version number to refresh the network topology.

The rank value in RPL reflects mainly the position of a given node relative to the sink node. All the nodes upon starting joining the DODAG calculate their Rank relative to the root node. The calculation of the rank depends of the used objective function; Objective Function Zero (OF0) or Minimum Rank with Hysteresis Objective Function (MRHOF). OF0 prioritizes selecting the ideal parent node based on minimizing the number of hops, while MRHOF opts for the most efficient route to the sink node by minimizing the Expected Transmission Count (ETX).

2.2. Preferred Parent Selection and switching in RPL

In RPL Contiki, the default OF is MRHOF, based on which the rank of a given node is based mainly on the path quality from a given node up to the sink node, it is calculated as follows:

$$\text{new_rank} = \text{base_rank} + \text{rank_increase} \quad (1)$$

Where:

- **base_rank**: is the rank of the preferred parent.
- **rank_increase**: represents the link metric associated to this preferred parent.

The preferred parent node is typically a neighboring node with a lower rank, indicating a higher position in the network's hierarchy. The selection of the preferred parent depends on the best path metric related to each neighbor node, where the neighbor that has the lower path metric is selected to be the preferred parent. The path metric can be defined as the sum of the links metrics between every two successive parents, from the node in question up to the sink. The value of the path metric of a neighbor node is related to the rank value advertised by this neighbor and the link metric value associated to it:

$$\text{Path_Metric} = \text{Candidate Parent's Rank} + \text{link metric} \quad (2)$$

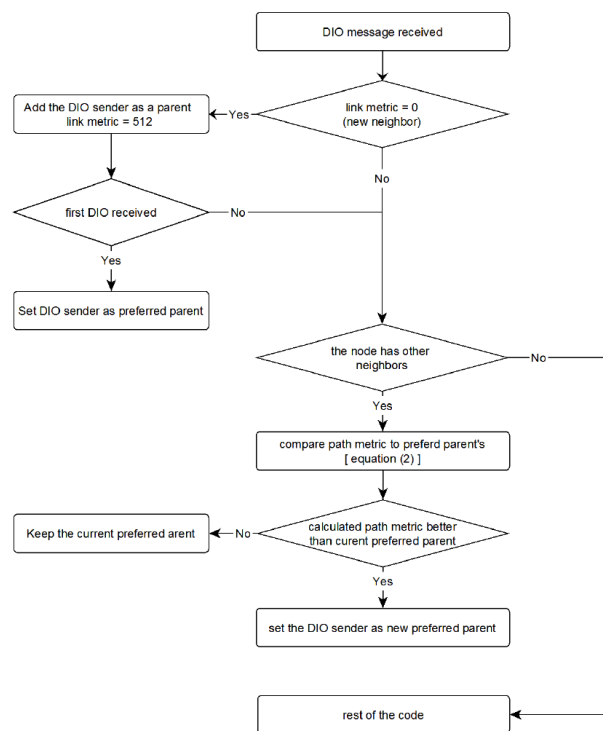


Figure 2. Preferred Parent Selection and Switching

Flowchart of figure 2 depicts how the preferred parent is selected and replaced, where the term “parent” is given to all neighbors, and the term “preferred parent” refers to the one selected preferred parent. Based on RPL Contiki principals, the first DIO sender in selected as preferred

parent. If the node does not have other neighbors other than the selected preferred parent, then the node has no need to calculate the path metric, since there is no second path to compare to. In the case of existence of other candidates, (receiving DIOs from other neighbors) the node must calculate the path metrics and compare them to its preferred parent's path metric. The node will replace its current preferred parent if one of the neighbors has better path metric, otherwise there will be no replacement of the preferred parent. Thus, a node switches its current preferred parent in a rank related matter in the case of receiving a DIO from a new neighbor with much better rank value, or in the case of an update in the rank of the current preferred parent, or in the rank of one of the nodes in the parents list.

2.3. ETX and Rank Calculation

The selection of the preferred parent is based on the best value of the path metric, calculated through each of the different candidates. The path metric through a given node presents the potential rank value the child node gets if the candidate parent is selected. Thus, a decisive parameter in the calculation of the rank is the rank increase which is the link metric, as depicted in (2). Based on Contiki, the link metric takes 512 as an initial value for every new neighbor. For an already known neighbor, the link metric is related to the calculated ETX.

The ETX is used to estimate the number of transmissions required for a successful packet delivery over a link. Higher ETX values indicate more transmission attempts due to unsuccessful packet deliveries, which can occur due to various factors such as collisions or node congestion. In Contiki's RPL implementation, each new neighbor starts with an initial ETX value of 512. For known neighbors, the ETX is updated based on the packet delivery performance. The ETX calculation occurs when a node sends a DAO (Destination Advertisement Object) message, either a regular DAO for the current preferred parent or a No-Path-DAO when replacing a parent. The ETX update depends on the packet transmission success. If a packet transmission fails (no acknowledgment is received), the ETX is set to a higher value (2560). For successful transmissions, the ETX is calculated using the number of attempts made, as follows:

$$\text{Packet ETX} = \begin{cases} \text{numtx} \times 256, & \text{Successful transmission} \\ 10 \times 256, & \text{Transmission fails} \end{cases} \quad (3)$$

- numtx represents the number of transmission attempts for a successful packet delivery. If a transmission fails, it takes the value 10.

The updated ETX for a link, referred to as the recorded ETX, combines the previous ETX value with the newly calculated packet ETX using a weighted moving average formula:

$$\text{New ETX} = \text{recorded ETX} \times 0.9 + \text{packet ETX} \times 0.1 \quad (4)$$

This formula 4 ensures that the ETX reflects recent transmission performance while still considering historical data. It helps to stabilize the ETX value by giving less weight to temporary transmission issues. For instance, if a node has experienced significant transmission failures, its ETX value will decrease gradually as the link shows improved stability. The calculated ETX value is then used as the link metric to influence the rank calculation and parent selection process, prioritizing more reliable links.

2.4. Impact of the Initial Link Metric Value

The link metric is always initialized at a value of 512 whatever the position of the DIO advertiser node in the topology. After getting a calculated link metric value that reflect the reel link quality,

the ETX is updated and consequently the path metric is updated as well through each neighbor. Based on the new reel calculated metrics, the node decided whether to keep its current preferred parent or to witched by a new one.

An initialized value is inevitable for all the nodes to starts the calculation of the ETX and get consequently the correct information about the quality and status of a link to a given neighbor. Based on our analysis, a large percentage of changing of the preferred parent is noticed because of this fixed initial link metric value. This feature leads to inaccurate decisions in choosing the preferred parent and to an instability in the topology. The changing in the preferred parents engenders several processes by the nodes, including:

- Resetting the trickle timer of DIO messages, that contain new updating information to rebuild routes,
- Sending a No-Path-DAO message to the replaced parent,
- Scheduling sending a DAO message,

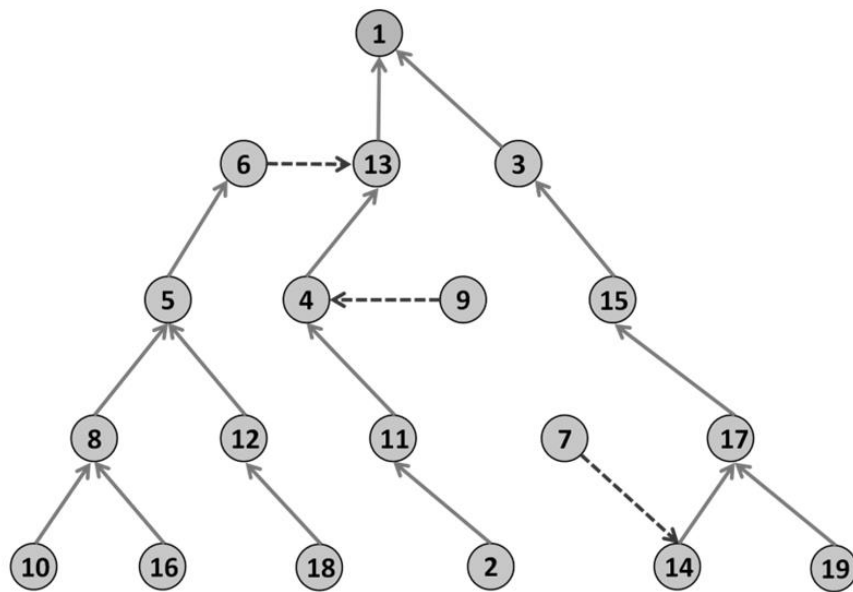


Figure3. Example of a probable topology

Figure 4 illustrates an example of a topology at an initial status. It can be seen that three cases present a must switch of the preferred parents.

- Node 6: in the case, node 6 had chosen a node with same hop count which is node 13 as preferred parent instead of the sink node, this can be due to the fact that it received the first DIO from node 13. After a while, node 6 will replace the preferred parent by the sink node. This switching operation engenders extra energy and control overhead consumptions as explained earlier, for the node its self and for the nodes of its Sub-Dodag.
- Node 9: in this case, an inevitable switching must take place, like for node 6. In this example the switching process engenders extra overhead exchange and energy consumption only for the node itself, since it has not children nods.
- Node 7: for this example, a probable case is treated, where the node 7 chooses a node with higher hops count; node 14. Here we have also an inevitable switching must take place, where the preferred parent must be switched to node 7 or node 17.

To address the previously detailed issue regarding the parents switching cadence, we propose a

International Journal of Computer Networks & Communications (IJCNC) Vol.17, No.1, January 2025
modification in the rank calculation method. Our contribution is based on giving the link metric a significant and meaningful value that reflects the position of the DIO sender node in the topology. This approach reduces parents switching because of the accurate calculated path metrics through the different candidate parents. On the other hand, the proposed mechanism eliminates the nodes with higher number of hops toward the sink to be selected as preferred parent. In our proposed modification we define two new variables:

N_Hop, recorded_etx_initialized and I_nbr_link_metric.

- N_Hop: presents the number of the hop of each neighbor compared with the Sink node.
- I_nbr_link_metric: is a link metric value related to the DIO sender node advertised rank, given

by:

$$I_nbr_link_metric = N_Hop * 2 * 128 \quad (5)$$

Where: 128 is the threshold of rank difference to switch a preferred parent defined by RPL Contiki

The flowchart below depicts our proposed modification:

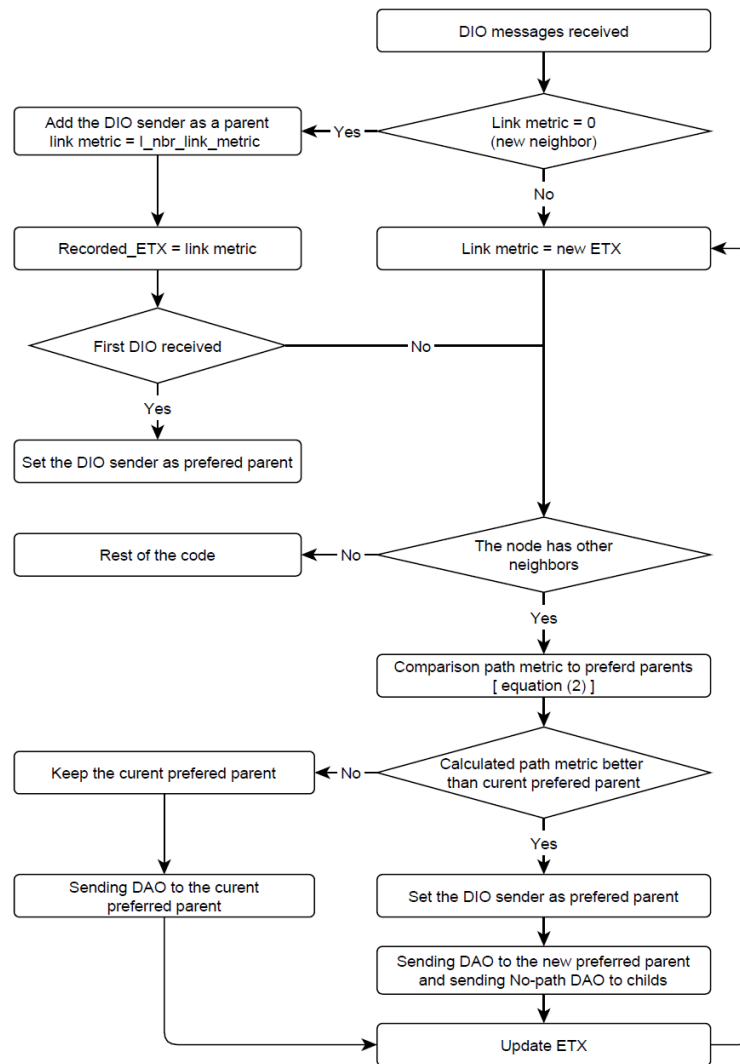


Figure 4. Flowchart depicts the proposed modification

3. RESULTS AND DISCUSSION

The simulations conducted in various IoT network scenarios involve a comparative analysis of three cases, each characterized by varying node densities. The simulations are executed using COOJA within the Contiki 3.0 operating system, a well-recognized OS commonly employed in the realm of IoT [17]. The simulations are conducted utilizing Z1 motes, the key attributes of which are concisely detailed in Table 1 [18]:

Table1.Simulation Parameters

Parameter	Value
OS	Contiki 3.0
Number of nodes	10,15,20,25,30
Power analysis	Power trace
μ Controller	MSP430F2617–16 bits RISC
CPU Frequency	16 MHz
Transceiver	CC2420 (IEEE.802.15.4), 2.4GHz

Z1mote	RAM	8KB with 92KB (flash memory)
	Voltage	3V
	TX Current	17.4 mA
	RX Current	18.8 mA
	LPM Current	0.426 mA
R Timer	32768 ticks per second	
Measurement interval	10 seconds	
Simulation Time	30 Mn	

The three simulated scenarios shown in Figure 5 involve a network comprising a single sink node with fluctuating numbers of nodes, specifically 25, 35, and 45.

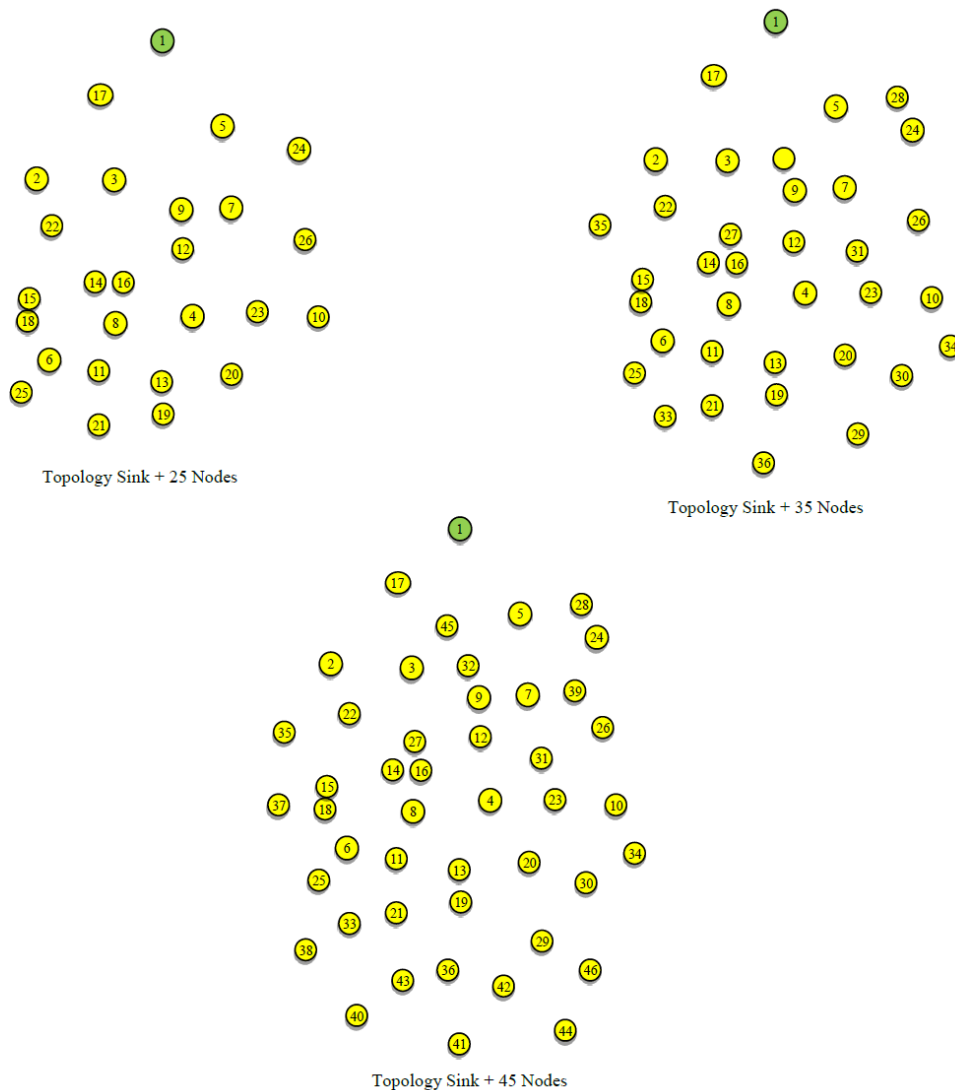


Figure5.The three studied topologies

Two parameters leading to changing the preferred parent are analyzed in figures 6. The first parameter is based on the initial link metric value given to each of the neighbors, and the second parameter is based on the ordinary update in the ETX calculation that leads to an update in the link metric value, between the node and each of its neighbors. These two parameters are analyzed in three simulated scenarios; topology with 25 nodes, topology with 35 nodes and topology with 45 nodes.

In the first scenario, we can notice that the default RPL engenders a total of 384 parent switching, 46.5% of it are due to the initial value attributed the link metric, while our proposed approach has engendered only 60 parents switching, where only 3 switching are because of the initial value given to the link metric. The same behavior is noticed in the second scenario, where 876 parent switching is recorded in default RPL, 42% of it are due to the initial value attributed the link metric, while our proposed approach has engendered much lower value of 115 parents switching, in which 15 switching are because of the initial value given to the link metric, where are recorded in the default RPL. In the third scenario the total for both cases increases, but the difference between the default RPL and the proposed approach gets lower, where a total of parents switching

International Journal of Computer Networks & Communications (IJCNC) Vol.17, No.1, January 2025
of 1401 and 729 are recorded in the default RPL and the proposed approach, respectively. These later results are due to the nature of the dense topology, where every node has higher number of preferred parent candidates with slight rank difference, which leads to a continuous parent switching. Based on the obtained results, our proposed approach succeeded at significantly reducing the recorded parent switching in the three scenarios, where the higher impact was noticed in the less dense topology, and the lower impact is noticed in the dense topology.

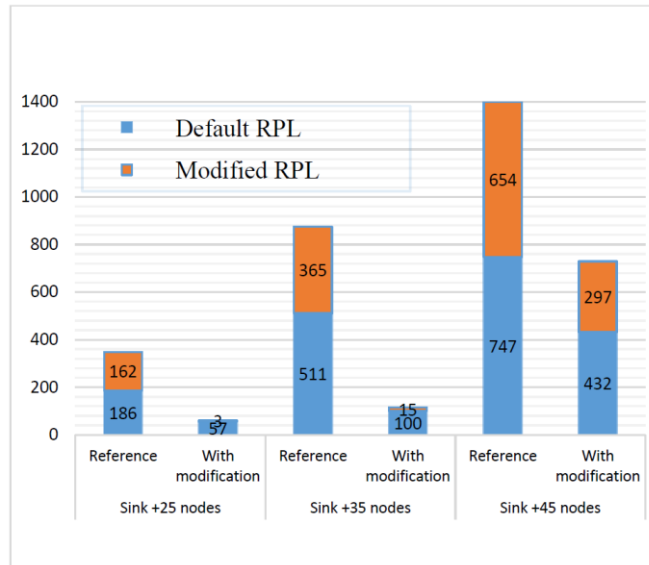


Figure 6. Statistics on changing the preferred parents

In order to highlight the impact of our work, we study the behavior of one specific node, node 11. Figure 7, illustrates the different nodes taken as preferred parents by node 11, in the reference network and after implementing the proposed approach. Regarding the reference network, node 11 kept switching its preferred parent from a list of 13 nodes, 5 of them are nodes with unacceptable rank value “higher rank value”. On the other hand, the proposed approach succeeded at narrowing the list of the nodes taken as preferred parents to 4 nodes, all of them with lower rank.

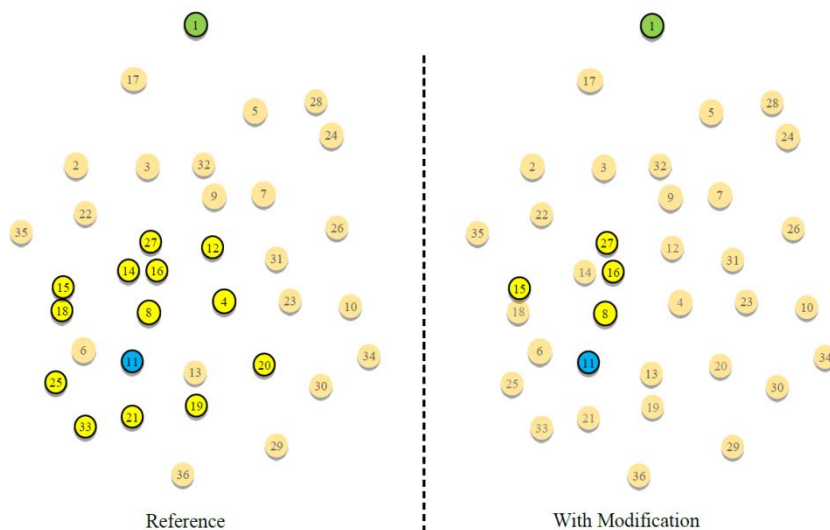


Figure 7. The positions of the preferred parents chosen by the node11

Figure 8 represents the chronology of the chosen preferred parent by node 11 during the 30 minutes of simulation, before and after implementing the proposed approach, in the 35 nodes scenario. In the reference topology, the node 11 reached a total of 28 changing of preferred parent from a list of 13 parent nodes, 13 out of the 28 changing are because of the initial inaccurate link metrics value attributed to the different neighbors, while the rest 16 switching are due to the updates in the ETX. Additionally, the figure shows that the first recorded 10 changes has been recorded just in the first 90s of simulation, 9 out of the recorded 10 switching are because of the initialized value of link metric. It's worth mentioning that the least period recorded during which a node is chosen as preferred parent by node 11 is 0.6 seconds, while the highest period was 9 seconds, these results reflect the suboptimal attributed link metrics initial values.

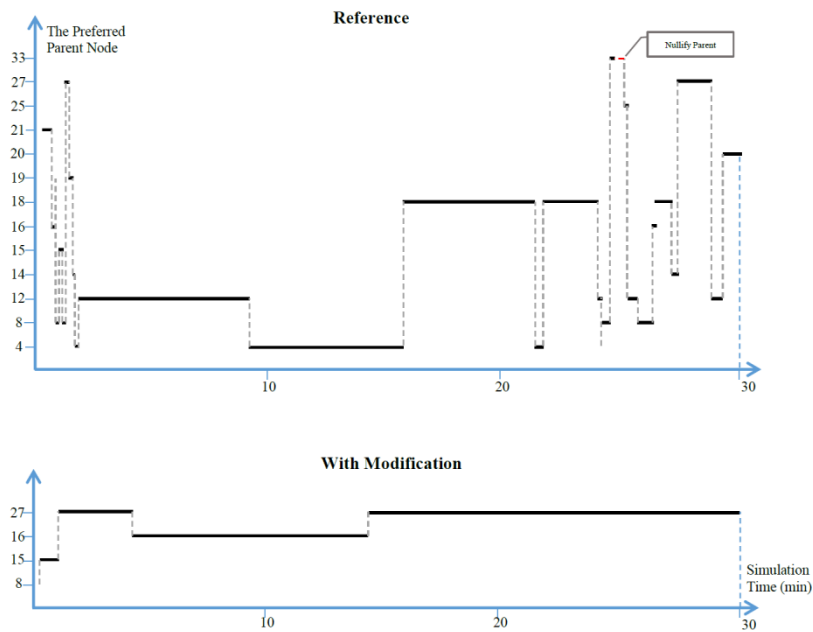


Figure 8. Chronology of the chosen Preferred Parent by the node11

After the first 90 seconds, an acceptable stability is noticed in the preferred parent switching, where the list of the chosen preferred parent was narrowed to nodes 12, 4 and 18, where all of them have lower acceptable rank. An interesting behavior is noticed after 24 minutes of simulation, where node 11 received a DIO from a new candidate parent “node 33”, consequently an initial link metric is attributed which led to its choice as preferred parent. Because of that initial attributed value, an instability and frequent parent switching has followed that event.

On the other hand, our proposed approach showed a noticeable stability in the topology, where node 11 had switched between only 4 candidates’ parents, which give an average of 7.5 minutes per parent. This result demonstrates the effectiveness of the proposed solution in reducing the useless parent switching within a given topology.

3.2. Control Traffic Overhead

Figure 9 summarizes the obtained results regarding the exchanged messages for the studied scenario. The number of generated and forwarded messages by the different nodes of the topology is proportional to the number of recorded parents switching, where the same behavior is noticed in the figure 8. The obtained results show that the proposed approach has succeeded at

International Journal of Computer Networks & Communications (IJCNC) Vol.17, No.1, January 2025
 significantly reducing the resulting message exchange, where the total has dropped from 3894 to 1752 in the case of the first scenario, and 60.5% in the total exchanged messages is recorded in the second scenario. In the third scenario, which is featured by a dense topology, 36.5% of reduction is achieved.

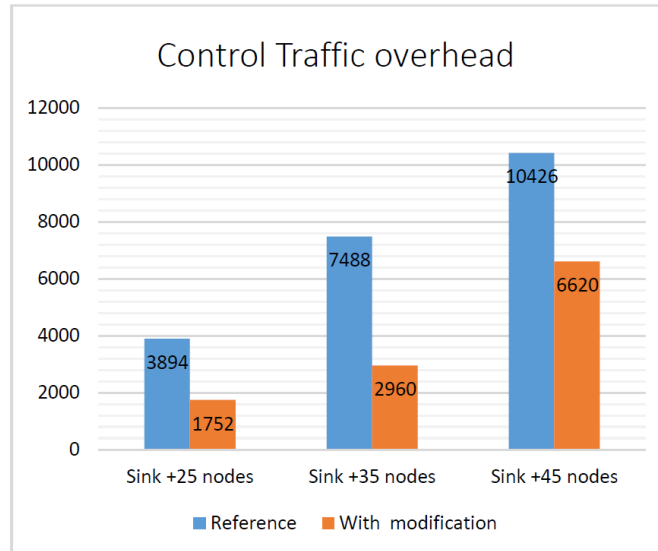


Figure 9. Control traffic overhead

3.3. Energy

Regarding the total consumed energy by the network's node, figure 10 illustrates the obtained results for the three scenarios with different densities. The same behavior noticed in the total exchanged overhead is also noticed in the case of the total consumed energy, where the density of the topology is inversely proportional to the impact of the proposed approach in reducing the consumed energy. The first scenario shows a reduction of 41.6%, while 37.3% of reduction is recorded in the third scenario.

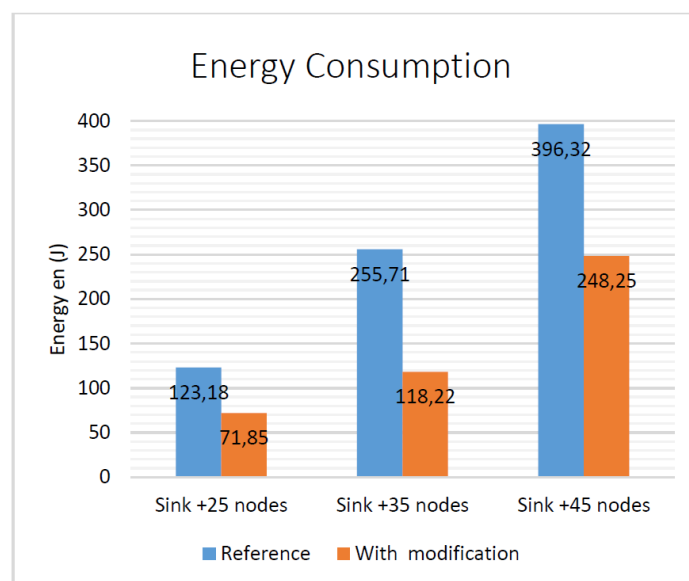


Figure10. The total energy consumed

3.4. PDR and Latency

The obtained results regarding the PDR and the latency are shown in the Figure 11. It depicts that whenever the density of network risen where the changes of preferred parent is more, it causes detrimental effects in PDR especially for the network 35 and 45 nodes, where the PDR decreases by an average of 50% and 71% respectively for the two densities. On other hand for our work the results confirm that its efficient in term of PDR where a noticeable improvement is noticed is this metric. Regarding the latency, we observe the efficiency of our proposed approach, and this is clearly in the two densities that detrimental effect where we notice a reducing by an average rate of 47% for the density 35 nodes and 81% for the density 45 nodes.

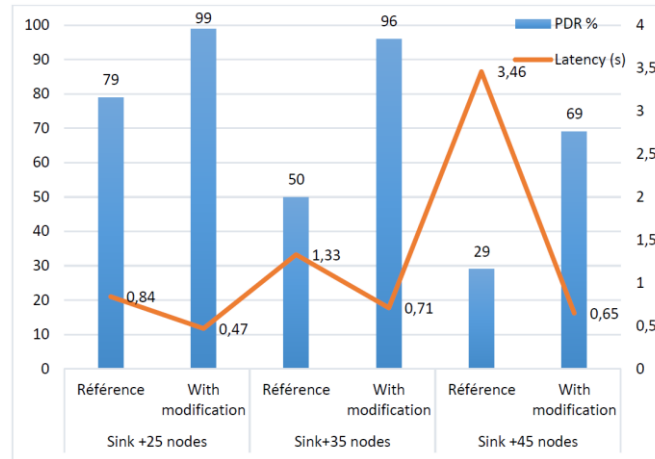


Figure11. Recorded PDR and Latency

4. CONCLUSION

In this work, we proposed a novel approach to enhance the lifetime of RPL-based IoT networks and optimize Quality of Service (QoS). Our approach reduces unnecessary parent switching by accurately assigning link metric values between nodes, ensuring a more stable and efficient network topology. Simulation results demonstrate the effectiveness of our method, showing a noticeable reduction in preferred parent changes, a decrease in control overhead, and in total energy consumption. Furthermore, the PDR improved and latency was reduced. These improvements contribute to network stability and reliability, ultimately extending the network's operational lifetime and enhancing overall performance.

REFERENCES

- [1] Nassereddine, M., & Khang, A. (2024). Applications of Internet of Things (IoT) in smart cities. In *Advanced IoT Technologies and Applications in the Industry 4.0 Digital Economy* (pp. 109-136). CRC Press.
- [2] Aishwarya, C. V. S., Caleb Joel Raj, J., Mandal, S. K., Reddy, C. N., & Mishra, B. (2022). Smart Health Care by Harnessing the Internet of Things (IoT): Applications, Challenges, and Future Aspects. In *IoT Based Smart Applications* (pp. 35-54). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-04524-0_3.
- [3] Raj, D., & Sagar, A. K. (2023, June). Vehicular Ad-hoc Networks: A Review on Applications and Security. In *International Conference on Advanced Communication and Intelligent Systems* (pp. 241-255). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-45124-9_19.
- [4] Vinodhini, M., Rajkumar, S., & Subramaniam, S. K. (2023). Real-time Internet of LoRa Things (IoLT)-based accident detection and prevention system in vehicular networks towards smart city. *International Journal of Communication Systems*. <https://doi.org/10.1002/dac.5692>.

- [5] Mowla, M. N., Mowla, N., Shah, A. S., Rabie, K., & Shongwe, T. (2023). Internet of Things and Wireless Sensor Networks for Smart Agriculture Applications-A Survey. IEEE Access. doi: 10.1109/ACCESS.2023.3346299.
- [6] Gour, B., & Maurya, J. P. (2023). Intelligent and Smart Agriculture System Using Cooperative Approach. In Intelligent Sensor Node-Based Systems (pp. 211-228). Apple Academic Press.
- [7] De, S., & Kar, A. K. (2023). Exploring IoT Applications in Industry 4.0—Insights from Review of Literature. IoT, Big Data and AI for Improving Quality of Everyday Life: Present and Future Challenges: IOT, Data Science and Artificial Intelligence Technologies, 15-38. https://doi.org/10.1007/978-3-031-35783-1_2.
- [8] Bankar, V. R., & Nandurkar, K. N. (2023). Implementation of IoT technology for quality improvement in an automotive industry. Materials Today: Proceedings. <https://doi.org/10.1016/j.matpr.2023.03.485>.
- [9] Macheso, P. S., & Thulu, F. G. D. (2023, April). Roles of Optical Fiber Sensors in the Internet of Things: Applications and Challenges. In International Conference on Soft Computing for Security Applications (pp. 923-933). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-99-3608-3_64.
- [10] Bhatia, J., Italiya, K., Jadeja, K., Kumhar, M., Chauhan, U., Tanwar, S., Bhavsar, M., Sharma, R., Manea, D.L., Verdes, M. and Raboaca, M.S., 2022. An overview of fog data analytics for IoT applications. Sensors, 23(1), p.199. <https://doi.org/10.3390/s23010199>.
- [11] Rehman, Z., Tariq, N., Moqurrah, S. A., Yoo, J., & Srivastava, G. (2024). Machine learning and internet of things applications in enterprise architectures: Solutions, challenges, and open issues. Expert Systems, 41(1), e13467. <https://doi.org/10.1111/exsy.13467>.
- [12] Poorana Senthikumar S , "Enhanced Objective Function ETX Metric in Routing Protocol for Low-Power and Lossy Networks (RPL) ", International Journal of Computer Networks And Applications, August 2022, DOI:10.22247/ijcna/2022/214507
- [13] A. Ouhab, T. Abreu, H. Slimani, and A. Mellouk, "Energy-efficient clustering and routing algorithm for large-scale SDN-based IoT monitoring," in ICC 2020 - 2020 IEEE International Conference on Communications (ICC), Jun. 2020, vol. 2020-June, pp. 1–6, doi: 10.1109/ICC40277.2020.9148659.
- [14] J. Nassar, M. Berthomé, J. Dubrulle, N. Gouvy, N. Mitton, and B. Quoitin, "Multiple instances QoS routing in RPL: Application to smart grids," Sensors (Switzerland), vol. 18, no. 8, pp. 1–16, 2018, doi: 10.3390/s18082472.
- [15] Rouissat, M., Belkheir, M., Belkhira, H. S. A., Hacene, S. B., Lorenz, P., & Bouziani, M. (2023). "A new lightweight decentralized mitigation solution against Version Number Attacks for IoT Networks". JUCS: Journal of Universal Computer Science, 29(2), 118–151.
- [16] Rouissat, M., Belkheir, M., Mokkadem.A , "Parent supervision lightweight solution against version number attacks for IoT networks", Journal or seething else!!! , February 2023, DOI: 10.21203/rs.3.rs-2605250/v1
- [17] Algahtani, T. Tryfonas and G. Oikonomou, "A Reference Implementation for RPL Attacks Using Contiki-NG and COOJA," 2021 17th International Conference on Distributed Computing in Sensor Systems (DCOSS), 2021, pp. 280-286, doi: 10.1109/DCOSS52077.2021.00053.
- [18] Belkheir.M, Rouissat.M, Boukhobza.M, Mokkadem.A, Bouziani.M, "A new lightweight solution against the version number attack in RPL-based IoT networks", 7th International Conference on Image and Signal Processing and their Applications (ISPA), May 2022, DOI:10.1109/ISPA54004.2022.9786370