SJSI

Research Article

Mobility-aware Routing for Low Power and Lossy Networks

Authors:
Alaa Althalji
Souheil Khawatmi
Bader Aldin Kassab

Affiliation:
The Department of Systems & Computer Networks of Faculty of Informatics_Aleppo University, Syria.

Submitted 28 Feb 2023
Accepted 03 Apr 2023
Correspondence:
bader.kassab@gmail.com

Abstract

With the development that took place in the field of devices and communications and the need to make life easier by using technology, the idea of connecting everything to the Internet was developed, which is the basic concept and goal of the Internet of Things so we can communicate and interact with people and other things like smart devices. Because smart devices are limited in power, memory and processing we need a new specific technology called IEEE 802.15.4 with low power consumption and bandwidth limitations, which relied on the RPL routing protocol, but it did not support mobile nodes, so it is a challenge to achieve it without causing overhead in low power and lossy network. A new control message was proposed to make RPL routing protocol able to support mobility by adapting to the change in the network topology quickly. A new control message proposed called 'HERE', which is sent by the mobile node when it moves. 'STOP' message is sent when it stops to tell the neighbor nodes, so they will be aware of the status of the mobile node, which help them to quickly reconnect to the network. The performance evaluation showed better results than previous studies because it increased the speed of reconnecting the node to the network after it moved, which helps to increase the packet delivery rate and reduce the latency .

Keywords: IoT, Mobile node, Routing, IEEE 802.15.4, RPL, LLN, 6LoWPAN.

Introduction

The Internet of Things (IoT) makes all smart devices connect to the Internet anywhere at any time to form the Internet of Things [1]. Smart devices are described to be low power and limited capability of processor and memory. These devices can be linked together to form a network without an infrastructure in which nodes act as a router. Smart devices use IEEE 802.15.4 [2] and routing protocol RPL [3], but RPL did not support mobile nodes, so many researchers worked to improve it, but their results showed more overhead because of sending more control messages in the networks. other researches results were not accurate because of depending on RSSI value (Received signal strength indication) to detect the mobility of nodes, which is affected by obstacles and interference. Our contribution is to improve RPL protocol by making it able to ensure the nodes aware of the mobile node movement, so the node can reconnect to the network as soon as possible and reduce the disconnection time without increasing the transmission rate of control messages. The proposed protocol is lightweight and suitable for devices with limited and restricted specifications because it uses new control message which is not periodically sent.

The rest of this paper is organized as following: The first section introduces the RPL routing protocols, then related works are mentioned with discussions about the challenges of the research to design an efficient LLNs routing protocol. After that the proposed control message is explained, followed by the extensive simulation, performance evaluation and interpretations. Finally, the paper is concluded.

RPL (Routing Protocol for Low Power and Lossy Networks(LLN)) :[٣]

In the network layer of the IoT protocol stack using 6LoWPAN technology, The RPL routing protocol was developed by (ROLL) group and was described in RFC (6550). RPL is suitable for fixed devices not the mobile devices. The root node serves as a gateway to the Internet for devices in the network. RPL organizes a topology as a Directed Acyclic Graph (DAG) that is partitioned into one or more Destination Oriented DAGs (DODAGs), one DODAG per root. It sends periodic DIO (DODAG Information Object) messages to inform nodes of its existence. It invites the neighbour nodes to call it, and in turn, each node that hears the DIO message and wants to communicate with root node will send a DAO (Destination Advertisement Object) message, then each node that communicates with parent node sends DIO messages to make the rest of the nodes connect and this process is repeated until all nodes are connected together, which is shown in Figure 1 .

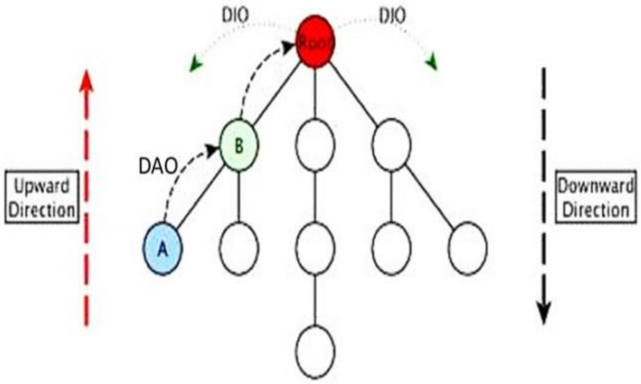


Fig. 1: Control messages used in RPL

The RPL protocol depends on the Trickle algorithm to manage the timers used to send periodic messages, to reduce the overhead on these networks. When the state of instability is detected, the transmission rate of control messages increases to spread updates quickly and reduces when the network is stable [4].

The node selects the parent node using Objective Function (OF) that depends on the number of hops, then it was developed to MARHOF (The Minimum Rank with Hysteresis Objective Function) [5] in which the parent node is selected based on the value of the Expected Transmission Count (ETX) that determine the quality of the link. The RPL protocol did not specify any mechanism for detecting routing adjacency failures of mobile node because such a mechanism causes overhead in bandwidth and power consumption. It is not suitable for Battery-powered devices to send periodic messages, so it requires external mechanisms to detect that the neighbour node is no longer reachable. This mechanism should preferably focus on links that were already used [٣] .

Related work

Fotouhi (2015) proposed the MRPL protocol by integrating the RPL protocol with the smart hop using beacons. The protocol has two phases: the discovery phase and the data transmission phase. In the route discovery phase, the mobile node sends a broadcast of n DIS messages. The node that receives DIS messages calculates the ARSSI and adds this value within a DIO message.

The mobile node selects its parent node that sent the DIO message with the highest ARSSI[٦]. Gara (2016) suggested to use adaptive timer algorithm to regulate the transmission of DIO and DIS messages by the mobile nodes. The proposed algorithm computes (d) the remaining distance for a node to leave the parent node's radio range by subtracting the preferred parent node's radio range from the value of the distance between the two nodes. When (d) distance becomes shorter, the node discovers to find a new parent node. The researcher suggested using ETX and RSSI values to determine the best parent node [٧].Cobârzan (2016) proposed Mobility-Triggered RPL (MT-RPL) is a cross-layer protocol operating between the MAC and routing layers. It enables X-Machiavel operations to use information from layer 2 to trigger actions at layer 3 to ensure the node connects to the network. It reduces the disconnection time, increases the packet delivery ratio, and reduces overhead. A restriction of MT-RPL is that it relies on a fixed node that acts as an opportunistic forwarder for packets sent by a mobile node [٨].Wang (2017) proposed an RRD (RSSI, Rank, and Dynamic) method to develop the RPL protocol based on a combination of Received Signal Strength Indicator (RSSI) monitoring, Rank updating, and dynamic control message. It proposes a new DIO interval by modifying it dynamically according to Rank updates. RRD increased the packet delivery ratio and decreased the end-to-end delay and overhead, but it did not consider energy consumption [٩]. Fotouhi (2017) proposed mRPL+ [10], when the link quality decreases during the transfer process with the parent node, the node will start sending periodic DIO messages to search for a better parent node. It also relied on the principle of overhearing to allow the parent node's neighbour nodes listen for all messages exchanged in their radio range. When a neighbour node detects good link quality with the mobile node, it sends a DIO message to link with it. mRPL+ achieved good results in terms of the packet delivery rate, reaching 100%, but it led to more power consumption· Bouaziz and Rachedi (2019) [11] focused on two phases in the protocol, the first one is motion detection for the nodes, and the second is predicting the new connection before the current connection is lost. It relied on the principle of excluding the mobile nodes from the path to avoid interruptions caused by them and considered the mobile nodes as a leaf in the network that isn't a router. EMA-RPL assumed that the node is moving away when the value of ARSSI decreases, but this is not always true because this value is affected by obstacles· Bouaziz (2019) [12] used the Kalman filter algorithm to predict the movement of mobile nodes and choose the parent node based on the predicted path. EKF-MRPL assumes that the mobile nodes will not participate in any routing process, but this is not always possible, especially in applications with many mobile nodes. Predicting movement and calculating distances are based on RSSI value which is inaccurate because it is affected by obstacles· Sanshi (2019) [13] modified the RPL protocol using fuzzy logic with several parameters (residual power, expected transfer count ETX, RSSI, mobility timer). FL-RPL uses the mobility timer parameter, which is the expected time to remain within the radio range depending on the location obtained from the RSSI value. This method isn't accurate because it is affected by barriers and interference. Mobile nodes are considered leaf nodes and cannot participate in the routing process, This concept is not correct in the case of a network with more mobile nodes than fixed ones· Manikannan (2020) [14] used the firefly algorithm (FA) inspired by the firefly that produces light to communicate or attract prey or as a protective warning mechanism. When the distance increases, the light becomes weaker. It depends on choosing the parent node with a high RSSI value and re-implementing the FA algorithm until it reaches an optimal solution. In the simulation, only 12 nodes were used, and FA-RPL improved the packet delivery rate by 2.31%. The firefly algorithm (FA) is a good and powerful optimization algorithm, but its computational complexity is high. Safaei (2022) [15]. The research proposed ARMOR protocol by using a new parameter TTR to select the best parent node that will stay the longest within the radio range. TTR is calculated based on the node's speed and position, and it is added to the DIO message. A new timer was added to increase the rate of sending DIO messages by the fixed node in order to introduce itself and to be selected as the parent node by the mobile nodes. The mobile nodes did not modify their timer, but this is not suitable for its neighbor nodes to be aware of their current speed in case it changes. The related works showed there was a need for a protocol that supports mobility, they worked on that but this led to an increase in delay and overhead in the network. In this research, new control message was proposed to make the nodes aware of parent node movement to take into account the appropriate time for changing the parent node without waiting for the timer specified in standard RPL protocol.

Table 1: Related works

Protocol	Proposed method	Disadvanteges
mRPL[6]	sent n DIS	Increased overhead
ARPL[7]	Used ETX, RSSI	RSSI effect with obstacles
MT-RPL[8]	Cross-layer protocol	Relies on opportunistic forwarder
RRD [9]	Used RSSI, Rank, and Dynamic	Did not consider energy consumption
mRPL+ [10]	Depend on overhearing	Increased power consumption
EMA-RPL[11]	Mobile node as leaf	Did not support mobility
EKF-MRPL [12]	Used kalman filter, RSSI	RSSI effect with obstacles
FL-RPL [13]	Used fuzzy logic, RSSI	RSSI effect with obstacles
FA-RPL [14]	Uesd firefly algorithm	computational complexity is high
ARMOR [15]	Used TTR parameter	Increase the rate of sending DIO messages

Proposed approach

To develop a mechanism for detecting routing adjacency failures caused by mobile nodes, a new control message (Here) is proposed. Here control message is an ICMP information message type about the state of the mobile node, consists of an ICMPv6 header followed by a message body, as illustrated in Figure 2.

ICMPv6 header

•Type: 8-bit. it indicates the type of message. The type value for the HERE control message is 200 because IANA has defined an ICMPv6 Type Number Registry, the values (0-127) are for error messages, while the range (128-255) is for information messages and the values 200 and 201 are reserved for private experimentation [16].

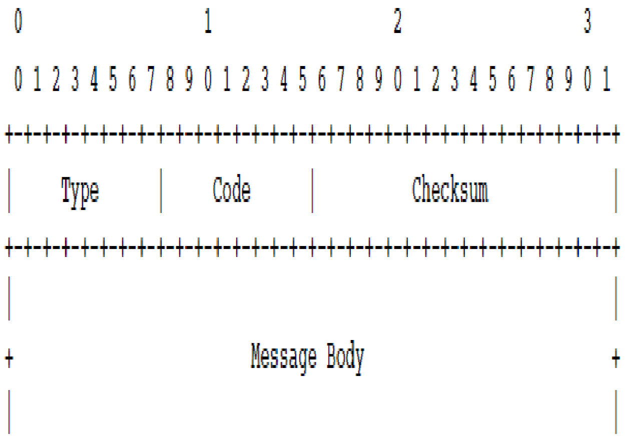
•(Code) depends on the type of message. It is used to create an additional level of message granularity [16]. Since RPL control messages take the following values .[٣]

- |x00 | DODAG Information Solicitation (DIS)
- |x01 | DODAG Information Object (DIO)
- |x02 | Destination Advertising Object (DAO)
- |x03 | DAO Acknowledgment (DAO - ACK).

So the suggested message HERE can be assigned a value of 4.

- |x04 | HERE

Checksum: The checksum field is used to detect data corruption in an ICMPv6 message [16].



```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                         Message Body                          +
|                                                               |
```

Fig. 2: ICMPv6 message format [16]

HERE Base object

HERE Base object is proposed to contain the following fields shown in Figure 3:

1- Flags: 8 bits. only 2 bits are reserved for S and H. the 6 bits remaining unused in the Flags field and reserved for flags. The field must be set to zero by the sender and must be ignored by the receiver .

(STOP) S: the 'S' flag indicates that the mobile node has stopped and sends a HERE message to its child nodes .

(LISTEN) L: the 'H' flag indicates that the node has heard Here message that is sent by mobile nodes and it is still within its radio range even after its movement, so the mobile node does not need to find a new parent node. However, if it moves and no such message arrives, the mobile node needs to find a new parent node as soon as possible to reduce the delay caused by separation resulted by moving the mobile modes when they change their place .

The values of all flag fields remain zero when a message is sent by the mobile node to its parent and child nodes if it moves to let them know.

(٠,٠)I MOVE TO HERE.

(٠,١)LISTEN I'M HERE

(١,٠)I STOP HERE.

(1,1) Invalid state.

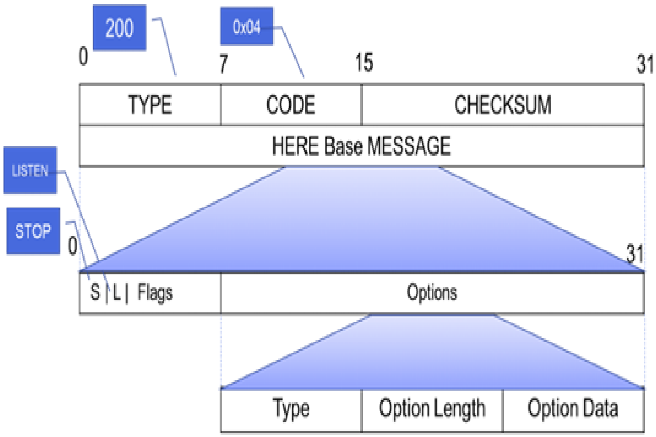


Fig. 3: HERE Base object

2- Control Message Option Format [16]:

No type of options was proposed, so padding will be used in proposed message. The general form of option contains three parts (Type, Option Length, Option Data). For padding option, the fields will be as following:

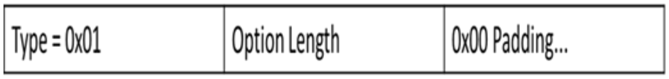
- Option Type: 8-bit, the value (0x00,0x01) is for padding type .
- Option Length: 8-bit, measured in octets, not including the Type and Length fields.
- Option Data: consists of N-2 zero octets.

The Pad1: option is used to add an octet of zeros for alignment. it has neither option length nor option data. The value of this type is 0x00. Pad1 is shown in Figure 4.

Type = 0x00

Fig. 4: Pad1

The PadN: option is used to add two or more octets of zeros. The value of this type is 0x01. PadN is shown in Figure 5.



Type = 0x01	Option Length	0x00 Padding...

Fig. 5: PadN

Proposed protocol rules

A mobile node must reconnect with a parent node within 5 seconds after it stops moving [17], so the following rules is suggested for proposed protocol:

•After sending a HERE message, a timer (HERE INTERVAL: 2.5 seconds) is triggered. If the

LISTEN message does not arrive, then the mobile node will search for a new parent node.

- A node that has received a HERE message from a parent node has a timer run for 2.5 seconds. If it does not receive another HERE message indicating that the node is moving within its domain, then it will need to search for a new parent node.
- A node that received a HERE message from a parent node and did not receive another message after 2.5 seconds then has to wait for 0.5 seconds. After this period (STOP INTERVAL: 3 seconds), if it does not receive a STOP message, indicating that the mobile node has stopped and is still within its range, then this detects that this node is out of its range and will need to search for a new parent node.
- After the node stops moving for 3 seconds, it sends a STOP message to tell the child node that it has stopped. If the child node hears this message, it will not search for a new parent node. Otherwise, it will search for a new parent node. The mechanism of the proposed protocol is described in Figure 6. When the mobile node moves, it sends the proposed control message 'HERE' to children and parent node. As the parent node receives this control message, it sends a 'STOP' control message indicating that it is still within its domain. Otherwise, it will search for a new parent node. The child node which receives 'HERE' control message waits for a time "HERE INTERVAL". After this period, if it does not receive another message, it has to wait for an additional time. If it receives a 'STOP' control message, this indicates that the node has stopped and is still in the same radio range. If the child node does not receive the STOP message, it will search for a new parent node.

The main purpose of this paper is to maintain the networks from disconnection for long time although of moving nodes, and to less changing parent node, so we make the node stay connected with the parent node although of its movement while it still in its radio range. The proposed control message is sent to inform the node of the change in the location of the parent node, which maintains communication between nodes by searching for the parent node immediately if it moves. The advantages of the proposed message compared to previous works that it is more suitable for LLN networks because it is sent only when node changes his location so it does not increase the overhead in the network.

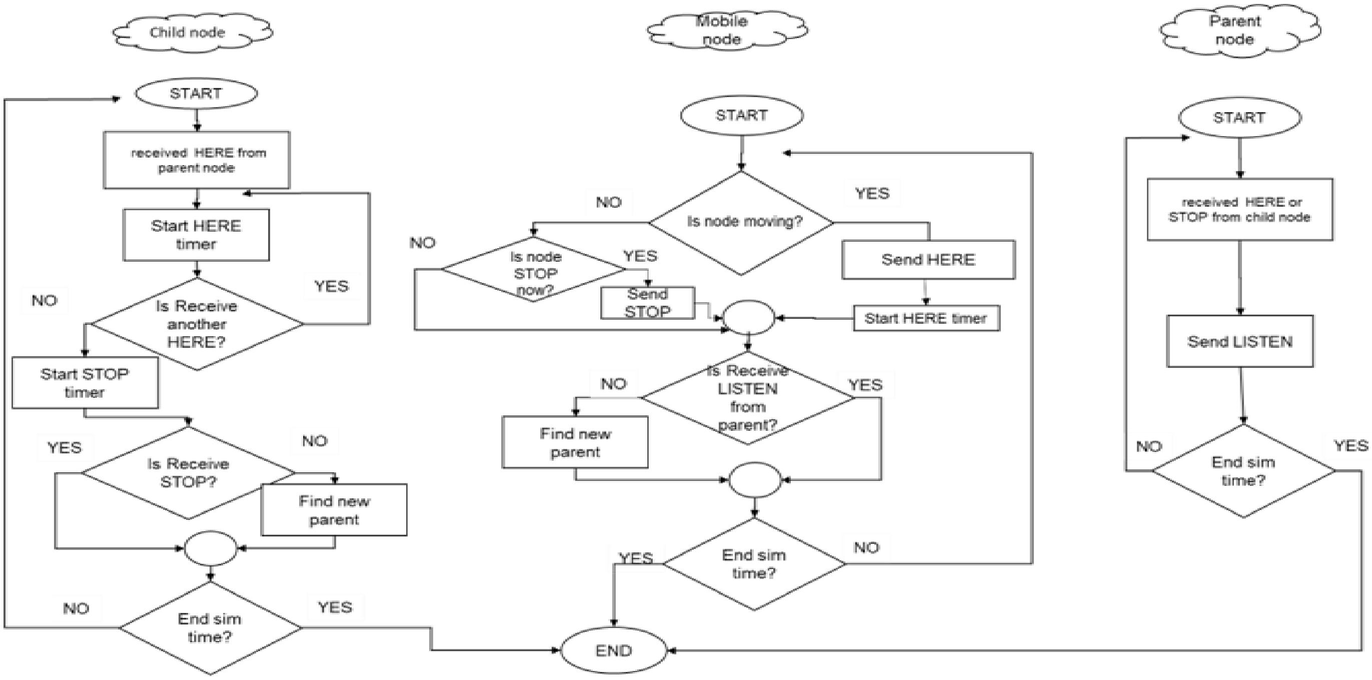


Fig. 6: The mechanism of proposed protocol

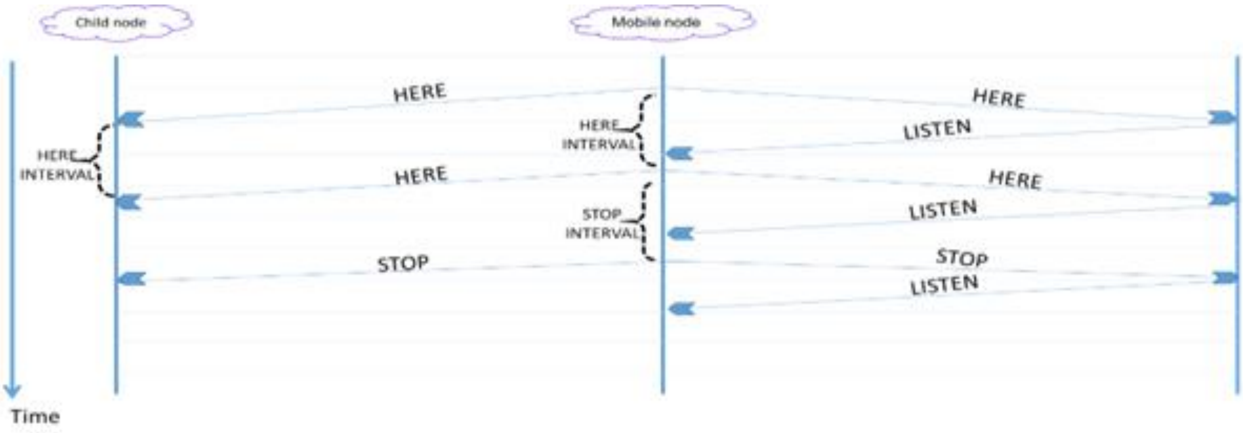


Fig. 7: Case study1

Figure 7 shows case study where all messages are received correctly despite of the mobile node movement. The parent and child nodes are still in the mobile node's radio range.

Figure 8 shows another case study where the mobile node was moving, which led the child node to exit outside the mobile node radio range, so it did not receive a HERE or STOP message. After that, it will search for a new parent node. When the mobile node stops, the parent node is out of range, so it does not receive a LISTEN message and it will search for a new parent node by sending DIS messages to its neighbours.

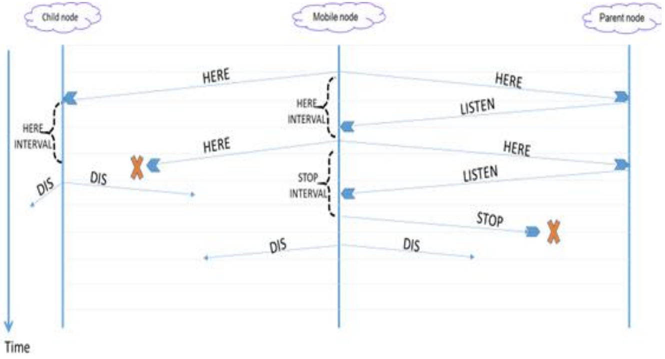


Fig. 8: Case study2

Protocol Performance Evaluation

In this paper, the proposed protocol was evaluated using the Cooja emulator [18] that supports IoT and all its protocols. Cooja is used because it is an emulator, not a simulator, meaning that its performance is closer to reality because it is running real devices in the network, which makes the results we get more accurate and simulating reality. This emulator runs on Contiki OS which is open source, multitasking and designed specifically for constrained memory. It supports a wide range of low-power wireless devices, like a Z1 chip or sky mote, etc .

[13] Performance metrics

The proposed protocol was evaluated in terms of PDR, power consumption, overhead, and end-to-end delay. The calculations are as follows:

1 .Packet Delivery Ratio (PDR): Represents the ratio between the number of received data packets and sent data packets.

2 .overhead: The ratio between the number of routing packets and the number of successfully received data packets.

3 .Average End-to-End Delay: The average time is taken to propagate the packet from the source to the destination.

4 .Average power consumption: is the average amount of power used by nodes during the working time of devices in the network.

Simulation Results and Analysis

This section presents a performance analysis of proposed protocols compared to protocols MARPL, FL-RPL, and ARMOR .The networks in the simulation were built using Cooja program, where the simulation parameters were adopted according to previous studies that were compared with .The research [19] suggested the MARPL protocol, which adopted the idea of detecting node movement through the value of RSSI and determining the availability of the neighboring node. If the node receives a DIO message, it updates the Neighbor Variability metric, otherwise, if it receives DAO or DIS control messages, it reduces the time interval between each transmission of DIO messages, which increases their transmission rate and thus more overhead. In the simulation, a 300*300 m2 area was considered with 50 mobile nodes at a maximum speed of 3 m/s with a different number of root nodes (1,2,3). The results in Figure 9 show the value of the packet delivery rate which increases when the number of roots nodes increases.

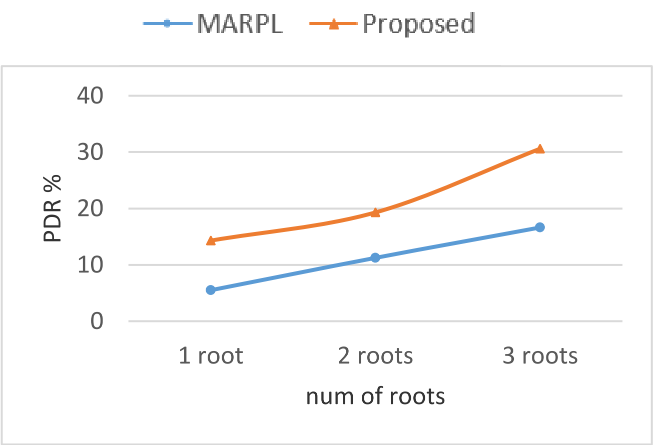


Fig. 9: Packet Delivery Ratio vs num of roots

By comparing the MARPL simulation results with the proposed protocol, we notice the superiority of the proposed protocol due to its ability to support mobile nodes by making the parent and child nodes of the mobile node aware of its movement via the proposed control message (HERE). MARPL increases the ratio of sending control messages, which causes more overhead on the network and increases collisions because it sends DIO messages to all neighbors, while the HERE control messages are sent only to the parent and child nodes that need this information. Average end-to-end delay: The result shown in Figure 10. Considering that MARPL relied on calculating the value of node availability through the RSSI, this leads to recalculating the value for all nodes every time. In the proposed protocol, even if the node moves and changes its location, it makes sure that the parent node is still within its radio range in order to reduce repeating the search process for a new parent node, and that reduces the power consumption, collision, and delay.

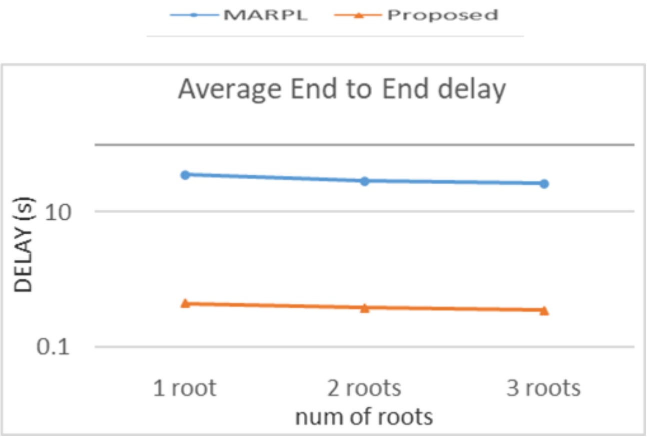


Fig.10: Average End to End delay vs num of roots

FL-RPL [13] is modified the RPL using fuzzy logic with several parameters. The simulation was implemented with an area of 10000^m2, 9 mobile nodes and (15,20,25,30) fixed nodes, the simulation results showed that the value of the packet delivery rate is close when comparing the proposed protocol with the FL-RPL protocol, where the proposed protocol exceeded 3% (Figure 11). The proposed protocol outperforms FL-RPL through reducing the delay by about half because the FL-RPL protocol performs many operations every time a node receives a DIO message, (see Figure 12). The routing metrics are given as input of fuzzy inference system to obtain a confidence score of the node and recalculate the mobility time. Therefore, these steps cause more delay and an increase in power consumption. Figure 13 shows that the proposed protocol reduced energy consumption, especially when the number of mobile nodes was more than fixed ones.

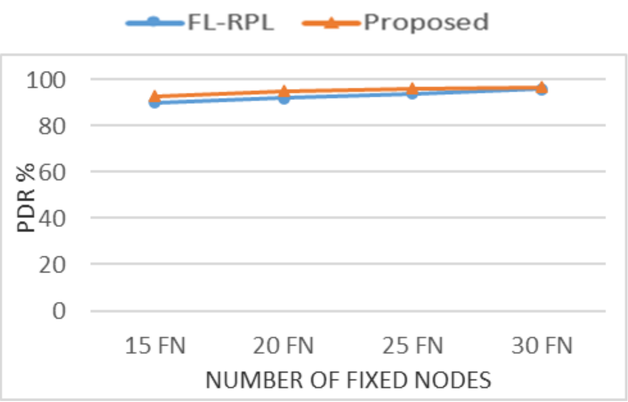


Fig. 11: Packet Delivery Ratio vs num of fixed nodes

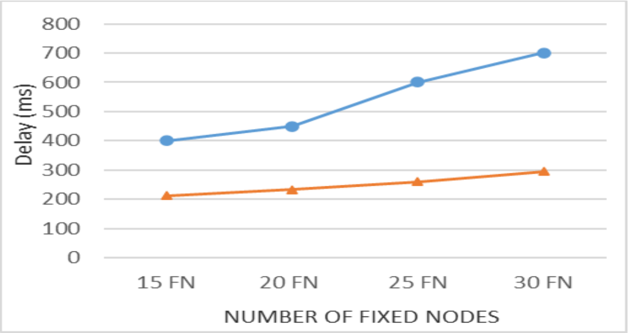


Fig. 12: Delay vs num of fixed nodes

ARMOR [15], the research proposed a new parameter TTR to select the best parent node that will stay the longest within the radio range. TTR is calculated based on the node's speed and position, and it is added to the DIO message. In this paper, a new timer was added to increase the rate of sending DIO messages by the fixed node in order to introduce itself and to be selected as the parent node by the mobile nodes. The mobile nodes did not modify their timer, but this is not suitable for its neighbor nodes to be aware of their current speed in case it changes. The simulation was implemented with an area of 10000^m2, 20 nodes(10 static nodes and 10 mobile nodes) at a speed of 0.5 to 1.5m/s, and one root node. Another scenario was with 40 nodes (20 static nodes and 20 mobile nodes).

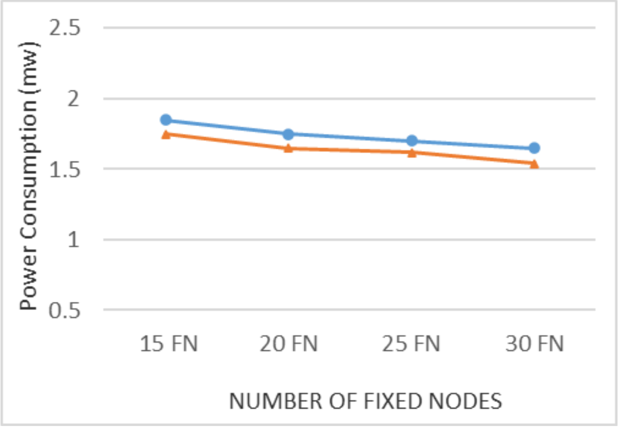


Fig. 13: Power Consumption vs num of fixed nodes

The simulation results showed that the packet delivery rate of the proposed protocol is 10% higher than ARMOR (Figure 14) because it supports mobile nodes by making them directly aware of the state of the parent node connected to them. If it becomes out of radio range, it will search for a new parent node.

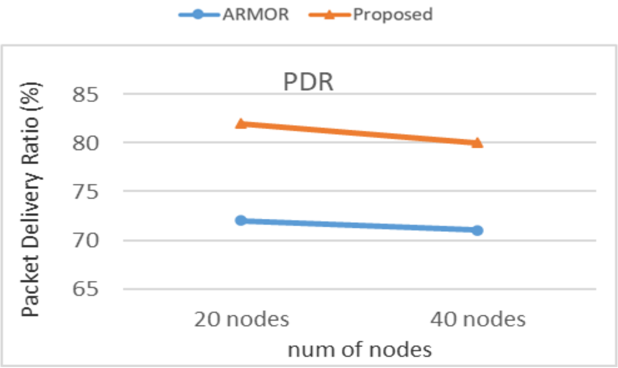


Fig. 14: Packet Delivery Ratio vs num of all nodes (mobile and fixed)

The routing load of the ARMOR protocol increased because it modified the timer algorithm for static nodes which made them send more control messages, so the mobile nodes are aware and communicate with them.

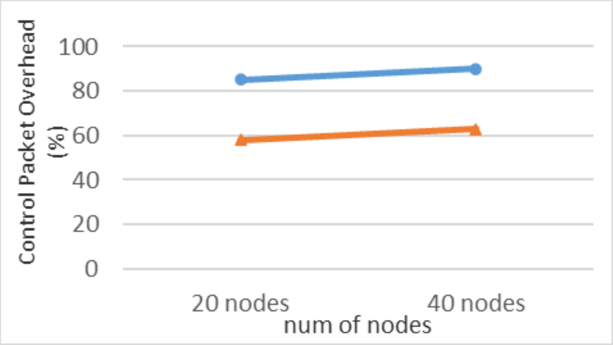


Fig. 15: overhead vs num of all nodes (mobile and fixed)

The proposed protocol did not increase the rate of sending control messages (Figure 15), so it was less routing load. It relied only on a suggested control message sent by the mobile node to its parent and child nodes when it moves. The power consumption of the ARMOR protocol is higher than the proposed protocol (Figure 16) because it sends more control messages

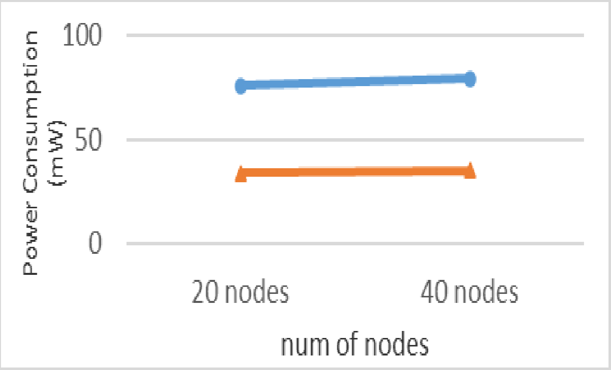


Fig. 16: Power consumption vs num of all nodes (mobile and fixed)

Discussion

The research shows the need to support mobile nodes in IoT networks. The proposed work helped to achieve this and reduce the impact caused by the nodes when they move within the network. From the simulation results, we observed that the proposed protocol improved the performance of the RPL protocol. It increased the packet delivery ratio because it made the parent and child nodes of the mobile node aware of its state. So they search for a new parent node immediately when the node moves away without waiting to expire the timer of the trickle algorithm, so it decreased the delay. The proposed protocol characterized that it doesn't send control messages periodically because this is not suitable for the nature of LLN networks. It minimized overhead because it maintained routing adjacency, focused on links that are used by the mobile node, and did not depend on broadcasting the proposed control message to all neighboring nodes to be more suitable for this type of network because this decreased the power consumption. Therefore, the proposed protocol helps to spread the devices that support mobility (smart watches, smart vacuum cleaners) in IoT networks without impact on the network.

Conclusions:

In this research, a new mechanism is proposed to discover disconnection in the network and makes a node reconnect as soon as possible. This disconnection results from movement of the nodes. Our goal is to ensure that the protocol is lightweight while working to support mobility, because many related studies led to increased the overhead, and others depended on discovering movement of nodes through the value of the signal strength RSSI, which is affected by interference and barriers. The new control message 'HERE' is proposed to be sent by the node when it moves to both the parent node and its children. If the parent node receives this message, it will respond by LISTEN message, but if it does not receive LISTEN message, it will search for a parent node. If a node stops moving, it sends a STOP message to notify its child nodes. These operations were set with a timer under the standard times for this type of network. The results showed the superiority of the proposed protocol over previous studies because it helped to reconnect the nodes in the network quickly, which increases the packet delivery rate and reduces the delay caused by disconnections in the network when nodes move. Also, it did not depend on increasing the rate of sending control messages which causes an increase in the network overhead. As this paper focuses on supporting the mobility in LLN networks, our future work is to propose a method to determine the type of device (fixed / mobile) and suggest a mechanism for detecting the movement of nodes in real-time within the network and changing its parent node to more appropriate node, so that it reduces the number of changes the parent node. In addition, we will work on enhancing network stability depending on additional parameters to choose the parent node in the network. This research contributes to increase IoT spreading in many applications in several areas, such as parking system, smart home, health care, etc. Which contain mobile nodes without affecting network performance.

References

1. **Pérez, T. and Camargo, J. Evolution of ubiquitous computing. Journal of Physics: Conference Series. 6th International Week of Science, Technology and Innovation; 2020 Nov 19-22; San Jose de Cucuta, Colombia; 2020.**
2. **Ott, A. Wireless Networking with IEEE 802.15.4 and 6LoWPAN. Embedded Linux Conference Europe. 2012 Nov 5-7; Barcelona, Spain; 2012.**
3. **Thuber, P. and Brandt, B. RFC 6550: IPv6 Routing Protocol for Low-Power and Lossy Networks. Internet Engineering Task Force (IETF) Request For Comments. 2008.**
4. **Aghaei, A. and Torkestani, J. LA-Trickle: A novel algorithm to reduce the convergence time of the wireless sensor networks. Computer Networks. 2021 Sep 4; 196.**
5. **Gnawali, O. RFC 6719: The Minimum Rank with Hysteresis Objective Function. Internet Engineering Task Force (IETF). 2012.**
6. **Fotouhi, H. and Moreira, D. mRPL: Boosting mobility in the Internet of Things. Ad Hoc Networks Elsevier. 2015 Mar 25; 26:17-35.**
7. **Gara, F. and Ben Saad, L. An adaptive timer for RPL to handle mobility in wireless sensor networks. International Wireless Communications and Mobile Computing Conference (IWCMC);2016 Sep 5-9; Paphos, Cyprus; 2016. p. 678-638.**
8. **Cobârzan, C. and Montavont, J. MT-RPL: a cross-layer approach for mobility support in RPL. EAI Endorsed Transactions on Internet of Things; 2016 Dec; 2(5).**
9. **Wang, J, Chalhoub, G. and Misson, M. Mobility support enhancement for RPL. International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN); 2017 Nov 28-30; Paris, France; 2017. p. 1-6.**
10. **Fotouhi, H. and Moreira, D. mRPL+: A mobility management framework in RPL/6LoWPAN. Computer Communications. 2017 Mar; 104:34-54.**
11. **Bouaziz, M. and Rachedi, A. EMA-RPL: Energy and mobility aware routing for the Internet of Mobile Things. Future Generation Computer Systems. 2019 Jan 12; 97:247-258.**
12. **Bouaziz, M. and Rachedi, A.EKF-MRPL: Advanced mobility support routing protocol for internet of mobile things: Movement prediction approach. Future Generation Computer Systems. 2019 Apr 15; 93:822-832.**
13. **Sanshi, S. and CD, J. Fuzzy optimised routing metric with mobility support for RPL. IET Communications. 2019 June 1; 13(9):1253-1261.**
14. **Manikannan, K. and Nagarajan, V. Optimized mobility management for RPL/6LoWPAN based IoT network architecture using the firefly algorithm. Journal Pre-proof. 2020 Sep; 77.**
15. **Safaei, B. and Mohammadsalehi, A. ARMOR: A Reliable and Mobility-Aware RPL for Mobile Internet of Things Infrastructures. IEEE Internet of Things Journal. 2022 Jan 15; 9(2):1503-1516.**
16. **Conta, A. Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification RFC4443. Network Working Group. 2006.**
17. **Martocci, J. RFC(5867) Building Automation Routing Requirements in Low-Power and Lossy Networks. Internet Engineering Task Force (IETF). 2010.**
18. **Bagula, B. and Erasmus, Z. IOT EMULATION WITH COOJA. department of computer science, university of the western cape (UWC). 2015.**
19. **Kniess, J. and de Figueiredo Marques, V. Mobility Aware RPL (MARPL): Mobility to RPL on Neighbor Variability. International Conference on Green Pervasive and Cloud Computing Springer; 2019 May 26-28; Uberlândia, Brazil; 2019. p. 59-73.**

Author contributions

Conceptualization: Alaa Althalji.
Methodology: Alaa Althalji.

Project administration: Dr. Souheil Khawatmi
Supervision: Dr. Souheil Khawatmi - Dr. Bader Aldin Kassab
Writing – original draft: Alaa Althalji.
Writing – review & editing: Dr. Souheil Khawatmi - Dr. Bader Aldin Kassab

Competing interests

Authors declare that they have no competing interests.

Data and materials availability

All data are available in the main text.