

## Article

# Impact of Router Count on Network Performance in OpenThread

Xaver Zak , Peter Brida  and Juraj Machaj 

Department of Multimedia and Information-Communication Technology, Faculty of Electrical Engineering and Information Technology, University of Zilina, 01026 Zilina, Slovakia; peter.brida@uniza.sk

\* Correspondence: xaver.zak@feit.uniza.sk (X.Z.); juraj.machaj@uniza.sk (J.M.)

## Abstract

A low-power IPv6 mesh standard, Thread, is gaining traction in smart-home, building-automation, and industrial IoT deployments. It extends mesh connectivity with the help of Router-Eligible End Devices (REEDs), which can be promoted to, or demoted from, the router status. Promotion and demotion hinge on two tunable parameters, the router upgrade and the router downgrade thresholds. Yet the OpenThread reference stack ships with fixed values (16/23) for these thresholds. This paper presents a systematic study of how these thresholds shape router-election dynamics across diverse traffic loads and network topologies. Leveraging an extended OpenThread Network Simulator, a sweep through both router upgrade and router downgrade thresholds with different gaps was performed. Results reveal that the default settings may over-provision routing capacity and may result in increased frame retransmissions, wasting airtime and reducing energy efficiency.

**Keywords:** OpenThread; Thread network; low-power IPv6 mesh; router selection

## 1. Introduction

The Internet-of-Things (IoT) is rapidly moving from isolated prototypes to large-scale, production-grade deployments that must remain reliable for years while running on severely energy- and memory-constrained hardware. To satisfy these constraints, many vendors have converged on Thread, a low-power IPv6 mesh protocol that builds on IEEE 802.15.4 and employs distributed routing to avoid single points of failure [1]. Thread's self-healing mesh is attractive for smart-home, building-automation, and industrial sensing applications, yet its performance depends critically on the number of devices that act as routers rather than end devices. On the one hand, a low number of routers may degrade connectivity, while on the other hand, too many routers could waste airtime and battery capacity.

Thread manages this trade-off through Router-Eligible End Devices (REEDs). Each REED can request promotion to a router when the mesh is sparse and demote back to an end device when routing capacity is plentiful. Two tunable parameters govern these transitions as follows: the router upgrade threshold (the maximum number of routers that still trigger a promotion) and the router downgrade threshold (the minimum number of routers that still trigger a demotion). These thresholds are configurable for each REED; however, the OpenThread reference implementation sets them to 16 and 23, respectively [2,3].

The Thread protocol stack has become quite popular lately for low-power IPv6 mesh network deployment. However, there is a lack of research papers and experiments that would validate the parameters of these networks. Overall, the Thread networks can achieve reliable data transfers with good energy efficiency while providing scalable topology [4].



Academic Editor: Chengxi Zhang

Received: 21 November 2025

Revised: 12 January 2026

Accepted: 16 January 2026

Published: 19 January 2026

**Copyright:** © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Thus, the OpenThread networks have great potential for implementation in Industrial IoT (IIoT) [4,5]. Nevertheless, it is important to understand how different settings would impact the performance of the networks.

The research involving OpenThread networks is mostly oriented toward possible security issues [6,7]. This is an important area, especially when the network is implemented in an IIoT environment. However, there are still open questions when it comes to the deployment of the OpenThread networks in an energy-efficient way. The OpenThread network devices come with predefined parameters, which may affect the energy consumption as well as data transfer efficiency. It is envisioned that modification of these parameters could improve the performance and efficiency of the OpenThread networks.

Mihaeljans and Skrastins [8] have performed simulations of the OpenThread network to analyze how much overhead is generated by the devices in the network, as well as the dependency of the generated overhead on the number of routers in the network. Moreover, they evaluate the connectivity when one or more routers are disconnected from the network.

Seehamat and Khunboa [9] studied the performance of the Constrained Application Protocol (CoAP) [10] in the OpenThread network in a real-world environment. Based on the results, the increased number of nodes can lead to network congestion. They achieved a packet loss rate of more than 50% when 150 packets were sent in the network with six nodes.

Khattak et. al [11] evaluated the performance of the Thread network in a real-world implementation in a smart city environment. They conclude that the choice of the maximum number of hops is crucial and depends on the application. While one-hop communication can provide very low latency, two-hop communication should be used in cases when increased coverage is desired, e.g., home automation and IIoT applications. Moreover, multi-hop communication is recommended for applications that demand high robustness and scalability. They were able to achieve low jitter and packet loss ratio in the OpenThread network using Raspberry Pi devices.

According to the above-mentioned works, it seems that in real-world implementations, some OpenThread networks may not perform optimally. Therefore, it is necessary to analyze the performance of the network under different scenarios and find parameters that have the potential to improve the network performance, i.e., reduce the packet loss rate and air time, in order to save energy of individual nodes.

This paper provides, to our knowledge, the first systematic exploration of how router upgrade and downgrade thresholds influence router-election dynamics in the OpenThread network. Using the OpenThread Network Simulator (OT-NS), this study evaluates the network performance under three scenarios. In each scenario, the router upgrade and downgrade thresholds are changed, and the impact on network performance is evaluated. In the first scenario, different hysteresis gaps (threshold gaps) are tested on a single topology. The second scenario evaluates different settings of thresholds across multiple topologies, i.e., grid layouts with different node spacings and a random node placement. And the third scenario is focused on the evaluation of different payload sizes. The main contributions of this paper are as follows:

- Investigation of the effects of router upgrade and downgrade thresholds on network performance (i.e., reliability of network and required transmission time), with multiple network topologies.
- Investigation of the effect of the gap between router upgrade and downgrade thresholds on the network performance.
- Investigation of the payload size impact on the network performance.

Based on the achieved results, it can be concluded that the default threshold pair (upgrade = 16, downgrade = 23) may not be universally optimal for all topologies and

that alternative settings can improve the network performance from both throughput and energy efficiency perspectives. The main idea is to reduce the energy consumption of nodes by reducing Tx time, which can be achieved by reducing the number of collisions, Tx retries, and multicast transmissions. Moreover, reducing the number of collisions will also improve the reliability and throughput of the network.

The remainder of the paper is organized as follows. Section 2 surveys the Thread protocol and device roles, as well as the attachment procedure that leads a REED to seek routing capacity. Section 3 describes the simulation environment and testing scenarios. Achieved results are presented and discussed in Section 4, and Section 5 concludes the paper and offers directions for future research.

## 2. Thread Protocol and Device Roles

Thread deliberately constrains the possible node implementations so that every device on the OpenThread network can be reasoned about in terms of two capability classes—Full Thread Devices (FTDs) and Minimal Thread Devices (MTDs). Alongside device type, there are three main roles in a network, as follows: leader, router, and child. Within a Thread partition, a single leader oversees the mesh's routers while collecting and propagating configuration data network-wide. The most reliable node should be elected as the partition leader via a leader election algorithm that uses a leader-weight value. Each node computes its own leader-weight value based on the device's power-supply class, border-router functionality, Commercial Commissioning Mode (CCM) capabilities, and operational stability. Moreover, an optional vendor-defined adjustment can further bias the metric of a particular node.

### 2.1. Full Thread Devices (FTD)

The primary distinction between Full Thread Devices (FTDs) and Minimal Thread Devices (MTDs) centers on energy management and multicast behavior. FTDs listen to the complete set of router multicast addresses and keep their radios continuously active (even while idle), thereby precluding the radio into sleep state. In contrast, MTDs may conserve energy by powering down their radios during the intervals between polling operations. Although every FTD listens to all router multicast addresses, only routers and router-eligible end devices (REEDs) actively undertake inter-device exchanges, such as child–parent solicitation procedures.

Within the FTD class, the following three functional subclasses are recognized: routers, router-eligible end devices (REEDs), and full end devices (FEDs). Both REEDs and FEDs participate in the topology exclusively as child nodes. However, a REED possesses the capability to elevate itself to router status when specified criteria are satisfied. An example would be when the router count in the relevant Thread partition drops below the minimum threshold, or when another end device explicitly requests the REED to become its parent (router) role. OpenThread device types are shown in Figure 1.

In contrast to a FED, which does not participate in the transmission of multicast messages, routers and REEDs listen to the same multicast addresses and maintain their routing tables. Despite the REED's inability to route traffic from other nodes, REEDs passively listen for routing information from neighboring routers and keep routing tables updated. The routing tables are thus readily available when REED is elevated to a router role. The key difference between router and REED is that REED is not participating in the periodic multicast transmission of Advertisement messages. These messages contain leader and routing information. However, both device types participate in the periodic transmission of multicast Announce messages propagating their Thread network at different radio channels. Since a REED transmits fewer multicast packets compared to a router, it

lowers overall network transmission. Thus, REEDs may have a positive effect on network congestion and radio interference.

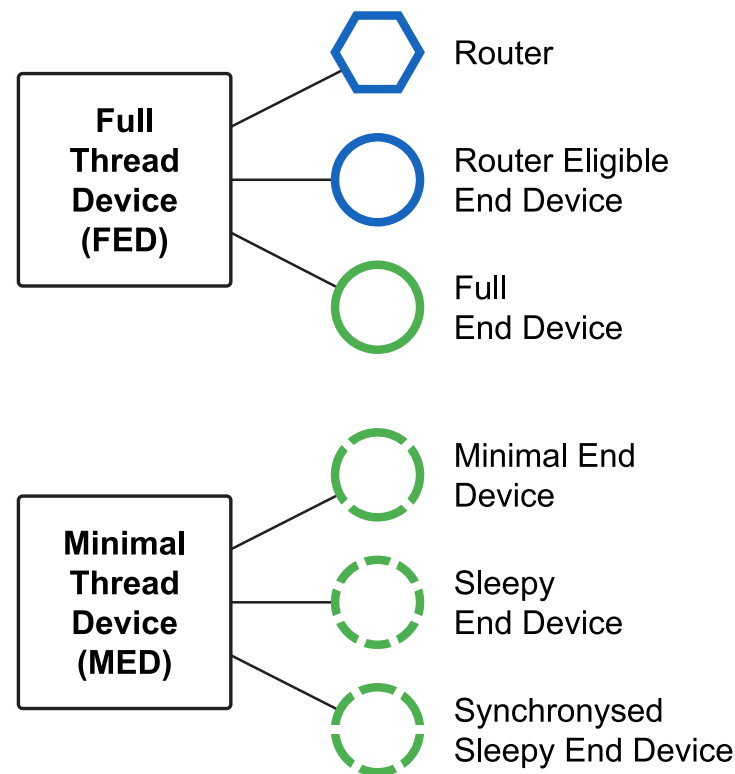


**Figure 1.** OpenThread device types.

## 2.2. Minimal Thread Devices (MTD)

Minimal Thread Devices (MTD) can only become a child in a Thread/OpenThread network. An MTD can be represented by three device types, i.e., Minimal End Devices (MEDs), Sleepy End Devices (SEDs), and Synchronised Sleepy End Devices (SSEDs). An MED maintains its transceiver in receive state when idle, thereby eliminating parent polling at the expense of constant current draw. On the other hand, an SED cyclically disables its radio and wakes at periodic intervals to poll for any buffered messages from its parent with a MAC Data Request. Similarly, an SSED, which was introduced in Thread version 1.2 [12], uses the Coordinated Sampled Listening (CSL) mode defined in IEEE 802.15.4-2015 [13].

Using the CSL mode synchronises parent transmissions with the child's sample period. Since the parent now knows exactly when the child will sample the medium, it schedules transmissions to align precisely with the child's receive window. During the Mesh-Link Establishment (MLE) attachment procedure, the SSED sets the CSL channel and timeout, and it then advertises its CSL period and CSL phase in the CSL Information Element (IE) of the IEEE 802.15.4 MAC header [14].

Therefore, the SSED simply wakes up, listens for any buffered messages from the parent, and immediately goes back to sleep. This eliminates unnecessary child data polling transmissions and minimizes receive windows. Thus, the key difference is that an SED must transmit a Data Request (poll request) every time it wakes up. In contrast, a SSED only has to listen, thereby eliminating those extra uplink messages and reducing both airtime and collisions, while extending battery life. Resynchronization happens on every data poll or SSED data transmission, or on demand when no traffic has been transmitted during the timeout period.

### 2.3. Device Attachment in a Thread Network

A detached Thread node sweeps every supported channel and transmits an IEEE 802.15.4 Beacon Request frame on each one. Routers and REEDs that receive this solicitation answer with link-local Beacons that enumerate their Personal Area Network ID (PAN ID), Extended Personal Area Network ID (XPAN ID), and Network Name. By iterating through all channels, the detached device builds a list of nearby Thread partitions, from which it may either select a target network or create a new partition if none are found and the device is capable of becoming a router [15].

Once the node possesses valid commissioning credentials for the chosen partition, it initiates the four-step attach procedure. Nodes always connect to the network as a Child, regardless of the capability to become a Router [15]. The four-step attach procedure goes as follows:

1. Detached node issues a multicast Parent Request MLE message that contains capabilities of detached node (Mode), Thread version that is currently running (Version), validation string for response timing of the parent message to mitigate replay-attack risks (Challenge), and scan mask. The scan mask specifies whether the response should be answered exclusively by routers or by both routers and REEDs. Parent requests can be sent multiple times with different scan masks and challenges to gather more data from neighboring routers and REEDs.
2. Every neighboring router or REED (if specified in scan mask) responds by sending a unicast Parent Response message to the detached node that echoes the Challenge string and supplies its RLOC16 (Routing Locator 16-bit) address, metrics such as Link Margin, Connectivity, Version, and current Leader data.
3. The attaching node ranks all parent candidates based on received data. After selecting its preferred parent, the child unicasts a Child ID Request message to its preferred parent. If the parent is a REED, it first promotes itself to Router, thereby extending the routing backbone, before processing the request.
4. The parent concludes the exchange with a Child ID Response that assigns the child's short RLOC16 address, conveys authoritative Network and Leader data, and establishes the inactivity timeout that governs future keep-alive traffic.

Upon receipt of the Child ID Response message, the node is formally attached to the Thread network as a Child. Any later elevation from Child to Router follows a separate delegation process.

### 2.4. Router Selection in Thread Network

The main task for routers in a Thread network is to establish a Connected Dominating Set (CDS), which has the following three criteria that must be fulfilled [16]:

1. A path between any two routers must be established exclusively through routers.
2. Every router can reach every other router while remaining entirely inside the router set.
3. Every End Device is linked to a Router without intermediaries.

A distributed algorithm continually adds or removes routers to maintain CDS and achieve minimum redundancy. Routers are added when the network holds fewer routers than the router upgrade threshold, requires additional path diversity, needs extra redundancy, or must expand capacity for more children. Conversely, routers are removed when the absolute maximum of 32 routers would be exceeded or when other network regions need spare router slots.

Every device joins the network as an End Device. If CDS conditions are not met, the REED may want to upgrade to router status. The upgrade procedure is permission-based and proceeds as follows:

1. Address Solicit—the REED unicasts an Address Solicit message to the Leader requesting a router-ID.
2. Leader decision—if the request is granted, the Leader replies with the assigned router-ID; otherwise, the REED remains a child.
3. Role transition—upon receipt of a valid router-ID, the REED reconfigures its state tables, assuming the router role.

Once the REED becomes a Router, it must integrate into the routing backbone by forming bidirectional links with its neighboring Routers. This is accomplished through Thread's MLE Link Request handshake. The freshly elected Router first multicasts a Link Request to all neighboring routers and REEDs, thereby announcing its presence and soliciting connectivity. Each Router that receives this solicitation replies with the following combined Link Accept and Request message: the "accept" portion acknowledges the prospective link, while the appended "request" portion solicits reciprocal information. Finally, the new Router answers to each responder individually with a unicast Link Accept, completing the handshake and establishing a pairwise Router–Router adjacency.

### 3. Simulation Environment and Scenarios

The OpenThread Network Simulator (OTNS) builds directly on Google's open-source OpenThread stack. Similar to the Radio Co-Processor (RCP) design, which allows the OpenThread core to be hosted on a host processor separate from the Thread radio, OTNS runs a dedicated POSIX process for every virtual node and simulates both the radio link and its surrounding environment. Every virtualized device runs in a separate process, with the IEEE 802.15.4 radio driver operating over sockets in the Linux kernel for communication. The end user can interact with the environment and devices via the Command line interface and the OTNS-Web graphical interface for real-time visualization and control [17–19].

In this paper, we rely on the OTNS version 2 [19], because it extends the previous version with the required research tooling as follows: the Key Performance Indicator (KPI) subsystem that exposes per-node counters and network-wide statistics, selectable RF-propagation models with run-time-tunable radio parameters, and more complex radio propagation models. Additionally, it allows the addition of multiple versions of OpenThread nodes.

Radio propagation model Mutual interference implemented in OTNS version 2 calculates the path loss of each transmitted frame on the receiver and adds a Bit Error Rate (BER) probabilistic model based on the Signal to Interference Ratio (SIR) between the received message and interference signals from other nodes. The path loss model is based on 3GPP TR 38.901 V17.0.0, which studies signal propagation models for frequencies in the range between 0.5 GHz and 100 GHz. The implemented Path loss  $PL_{\text{InH-NLOS}}$  House/Office model defines attenuation in dB units as follows [20]:

$$\begin{aligned} PL_{\text{InH-LOS}} &= 32.4 + 17.3 \log_{10}(d_{3D}) + 20 \log_{10}(f_c), \\ PL'_{\text{InH-NLOS}} &= 38.3 \log_{10}(d_{3D}) + 17.30 + 24.9 \log_{10}(f_c), \\ PL_{\text{InH-NLOS}} &= \max(PL_{\text{InH-LOS}}, PL'_{\text{InH-NLOS}}), \end{aligned}$$

where  $f_c$  denotes the center frequency in GHz, and  $d_{3D}$  is the 3D distance between transmitter and receiver in meters.

For each transmitted frame, shadow fading and time variant fading are calculated and applied. Additionally, based on transmission, SIR is recalculated based on the interference signals received from all other nodes. At last, the BER probabilistic model [21],

$$BER = \frac{8}{15} \times \frac{1}{16} \times \sum_{k=2}^{16} (-1)^k \binom{16}{k} e^{(20 \times \text{SINR} \times (\frac{1}{k} - 1))},$$

is applied to calculate if the packet will be successfully received. The same BER model calculation is used in the NS-3 LR-WPAN model [22,23].

### Testing Scenarios

We structured the study in OTNS around three progressively targeted scenarios. In the first scenario, we built a controlled  $6 \times 5$  grid of 30 Router-Eligible End Devices (REEDs) with 10 m spacing, forming a  $50 \text{ m} \times 40 \text{ m}$  rectangle. A single Border Router (BR) was placed above the grid, horizontally aligned with the center column. Generated traffic consisted of 50 Bytes (B) CoAP messages emitted every second from randomly selected REEDs to the BR. We varied the hysteresis gap between the router upgrade and downgrade thresholds, specifically 1, 3, 5, 7, and 10. For each hysteresis setting, the router upgrade threshold was swept from 1 to 31 to observe how hysteresis depth influenced routing behavior and overall network stability in this baseline topology.

The second scenario examined spatial effects. We created three grid topologies with identical structure but larger node spacings of 10 m, 20 m, and 30 m, and a topology with REEDs randomly distributed in the same area envelope as 10 m gap spacing ( $50 \text{ m} \times 40 \text{ m}$ ). Traffic again used CoAP messages with a 50 B payload every second from random REEDs to the BR. Here, the fixed hysteresis (threshold gap) was used and set to 3. Our analysis focused on how spacing and layout affected network performance and the number of routers while sweeping the router upgrade threshold.

In the third scenario, the 10 m grid spacing of 30 REEDs, i.e., the same topology as the first scenario, was used to explore traffic load sensitivity. In the simulations, CoAP messages with payloads of 10 B, 50 B, and 100 B were sent every second by randomly chosen REEDs to the BR. Additionally, no additional traffic was generated, and only OpenThread management was present. The router upgrade/downgrade threshold gap was set to 3. The router upgrade threshold varied from 1 to 31 to study its interaction with payload size. This design allowed us to isolate how application-layer traffic intensity and message size influence router formation and overall network behavior under identical spatial conditions.

Topologies used in individual scenarios are shown in Figure 2, where grid topology (a) is used with node spacings of 10 m, 20 m, and 30 m, i.e.,  $N = [1, 2, 3]$ . In each scenario, KPI parameters were collected across different sets of router upgrade and downgrade thresholds set according to the router upgrade threshold and threshold gap. For each set of thresholds, the network had a 10 min time period to converge as a conservative assessment. This period was used to allow for initialization of the network and election of routers without any further communication between devices. It allows us to analyze the selected parameters on a converged network without taking into account the traffic required to establish the network and for the initial setup of routers. After that, KPI measurements were performed for 60 min of simulated network traffic. In this period, network traffic was generated according to the scenario.

Each KPI measurement period was repeated 20 times. After data collection, the whole network was destroyed, and a new network was created with different router upgrade and downgrade thresholds. The flow of simulation is shown in Figure 3. For each router upgrade threshold sweep, we collected 620 samples of KPI measurements.

In the performed simulations, all the nodes except the Border Router were configured as REEDs. The simulation parameters, software, and hardware setup used in all scenarios are summarized in Table 1.

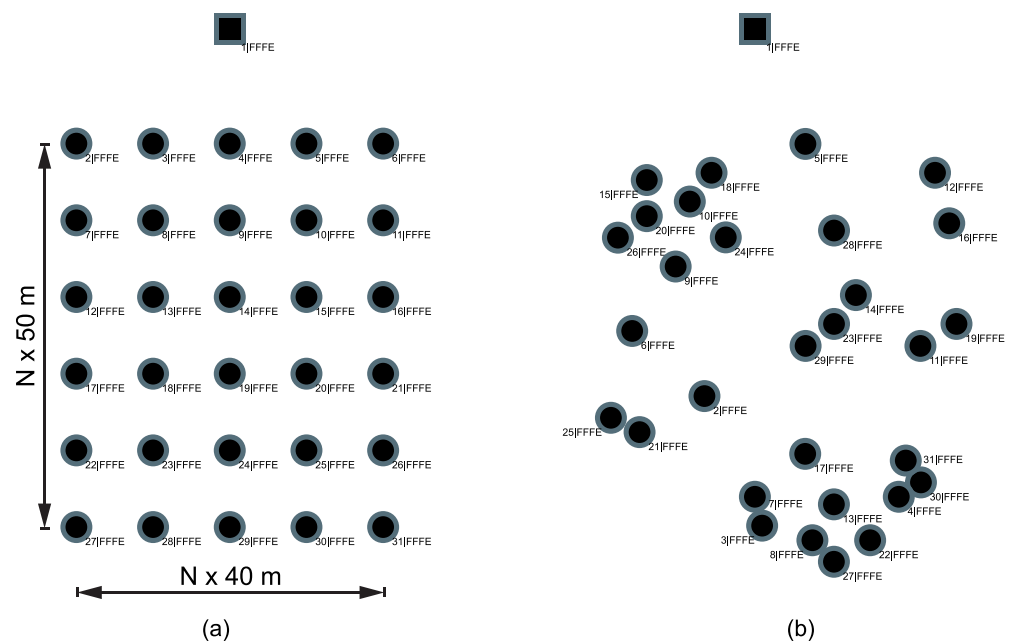


**Figure 2.** Network topology: (a) grid topology; (b) randomly distributed nodes.

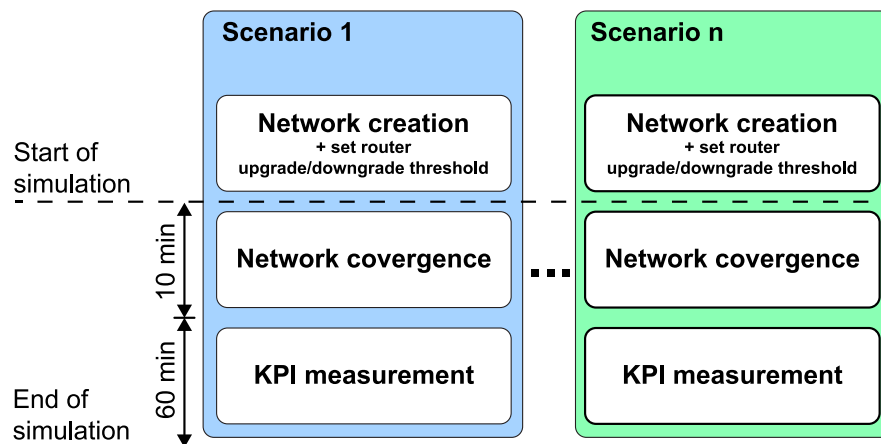


**Figure 3.** Simulation procedure.

In Scenarios where Constrained Application Protocol (CoAP) messages were sent with predefined 10 B, 50 B, or 100 B payloads, each CoAP message, defined in [24], may differ in the actual size of Bytes transmitted over the radio channel. There are two variable factors for that, as follows:

- Direct/over the mesh transmission of the message.
- Message fragmentation requirement.

OpenThread uses 802.15.4-2006 frame structure for data transmission, in which the maximal transmission unit (MTU) is 127 B [21]. On top of that, each physical frame transmits a 5-octet synchronization header and a 1-octet PHY header. Figure 4 shows the structure of the MAC, more precisely, how our CoAP payload is encapsulated in the MAC frame. Please note that the actual representation shows what is transmitted over the radio channel and does not show the CoAP message as a User Datagram Protocol (UDP)

datagram payload. It is encapsulated in an IPv6 packet since a UDP datagram and an IPv6 packet are reconstructed from a 6LoWPAN packet.

**Table 1.** Overview of simulation parameters.

| Hardware and software setup                |                     |
|--------------------------------------------|---------------------|
| CPU                                        | Intel Core i7-10700 |
| OS                                         | Debian 12           |
| Python version                             | 3.11.2              |
| Go lang version                            | 1.19.8              |
| OTNS version                               | Commit 76d2f19      |
| Radio model simulation parameters          |                     |
| Noise floor level                          | −95 dBm             |
| Standard deviation for shadow fading       | 8.03                |
| Standard deviation for time variant fading | 4                   |
| OTNS simulation seed                       | 1                   |
| Parameters of nodes                        |                     |
| OpenThread version                         | 1.4                 |
| Max number of frame retries                | 15                  |
| Receiver sensitivity                       | −100 dBm            |
| Clear channel assessment                   | −75 dBm             |
| Clock drift                                | 0 PPM               |
| Transmission power                         | 0 dBm               |
| Router selection jitter                    | 1                   |
| BR leader-weight                           | 65                  |
| REED leader-weight                         | 64                  |
| CSMA backoff period (first Tx)             | up to 2.24 ms       |
| CSMA backoff period (retransmissions)      | up to 9.92 ms       |

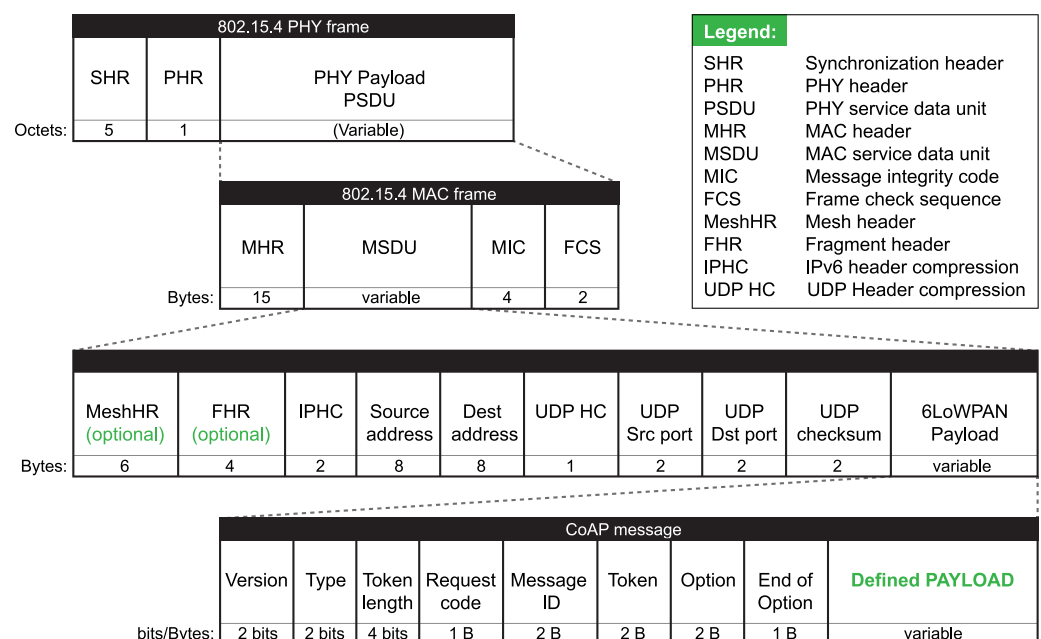


**Figure 4.** Content of MAC frame.

The MAC frame consists of a 15 B MAC header, a 4 B message integrity code, a 2 B frame check sequence, and a variable-length MAC service data unit (MSDU), which encapsulates the 6LoWPAN packet. The OpenThread uses a 6LoWPAN packet, which incorporates Improved Header Compression (IPHC) for the header compression of IPv6

packets and Next Header Compression for the header compression of UDP datagrams [25]. With those mechanisms, UDP datagrams and IPv6 packets can be reconstructed.

An optional 6 B Mesh header is added at the beginning of the 6LoWPAN packet in case the multi-hop transmission of the message is required. Because the minimum MTU of an IPv6 packet is 1280 B [26], a fragmentation and reassembly mechanism is needed [25], therefore the 4 B fragmentation header (FHR) will be added if the whole packet is larger than the available MSDU of the 802.15.4 MAC frame. In case of subsequent fragments of packet 5 B, FHR is used instead of 4 B. In our case, if the 6LoWPAN payload (CoAP message) exceeds the size of 81 B in direct transmission or 75 B in the case of the multi-hop transmission, fragmentation of the packet will occur. Please note that subsequent 6LoWPAN fragments, illustrated in Figure 5, only contain 5 B FHR, optional mesh header, and the remaining payload data.

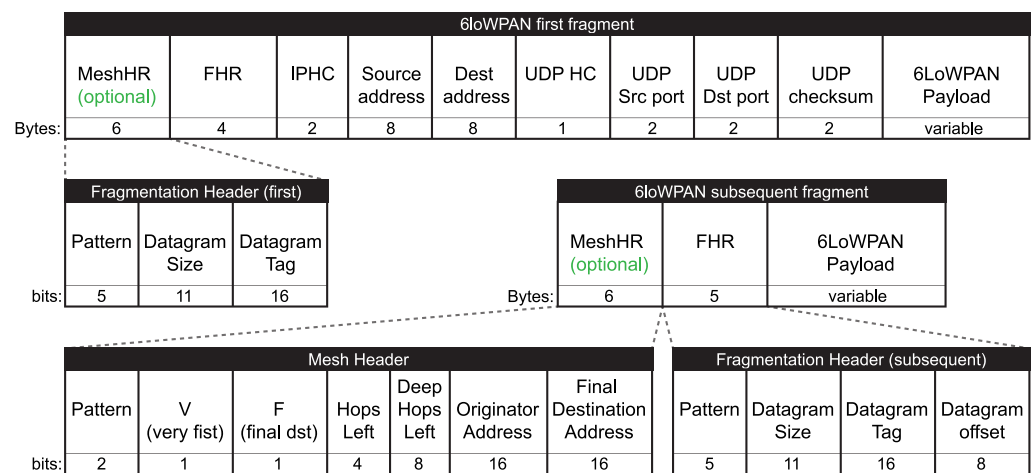


**Figure 5.** 6LoWPAN fragmentation headers.

Because of different header sizes in 6LoWPAN packets, the overall size of the MAC frame may vary. Table 2 shows MAC frame sizes and their respective transmission times based on different CoAP message payload (10 B, 50 B, and 100 B). Please note that the transmission times are calculated from the 802.15.4 PHY frame, which incorporates a 5-octet synchronization header and a 1-octet PHY header, where the transmission time for 1 octet is 32  $\mu$ s. Different sizes of MAC frames between multi-hop and direct transmission for 10 B and 50 B CoAP payload sizes are 6 B due to the added Mesh header. In the case of a 100 B CoAP payload size, there is a 12 B difference due to two packet fragments, where each contains the 6 B mesh header.

**Table 2.** Size of transmitted MAC frames in Bytes and Tx time.

| CoAP Payload [B] | Transmission Type  |                       |                    |                       |
|------------------|--------------------|-----------------------|--------------------|-----------------------|
|                  | Direct             |                       | Multi-Hop          |                       |
|                  | MAC Frame Size [B] | Tx Time [ $\mu$ s]    | MAC Frame Size [B] | Tx Time [ $\mu$ s]    |
| 10               | 65                 | 2272                  | 71                 | 2464                  |
| 50               | 105                | 3552                  | 111                | 3744                  |
| 100              | 122 + 63           | 6304<br>(4096 + 2208) | 120 + 77           | 6688<br>(4032 + 2656) |

## 4. Achieved Results

For each simulation scenario, multiple KPI parameters were periodically collected. The KPIs collected include the following:

- Mean number of transmitted frames in the network.
- Mean number of frames discarded due to frame check sequence (FCS) error.
- Mean number of frame retransmissions in the network.
- Mean transmission time in the network.
- Cumulative role changes in network.
- Mean CoAP message latency.
- Mean number of routers.

### 4.1. Scenario 1: Impact of Threshold Gap

In the first scenario, the impact of different gaps between the router upgrade threshold and the router downgrade threshold was analyzed. We simulated a  $5 \times 6$  grid topology with a distance between nodes set to 10 m. The simulation was run with five different threshold gaps as follows: 1, 3, 5, 7, and 10. A smaller gap (e.g., 1 and 3) represents small hysteresis, meaning routers should react quickly to changes. However, this may result in more router changes over the period of simulation. A larger gap (e.g., 10) introduces stronger hysteresis, meaning active routers in the Thread network should demote themselves less frequently and minimize role changes. In our study, we simulated 60 min of network traffic where a randomly selected node sends 50 B CoAP POST non-confirmable messages to the Border router every second. Each simulation with set router upgrade and downgrade thresholds according to the threshold gap was repeated 20 times.

Figure 6 shows the relation between the mean number of transmitted frames from all nodes and the router upgrade threshold. From the figure, it can be seen that the higher the routing upgrade threshold, the higher the mean number of transmitted frames in the whole network, despite different router upgrade and downgrade hysteresis. Major fluctuations in total transmitted frames can be observed when the router upgrade threshold is set between 2 and 5. A possible reason for this might be an insufficient number of active routers in the network, which can be observed in Figure 7, where the ratio between frame retransmissions and total transmitted frames is shown. It can be seen that a retransmission ratio of more than 10% occurs in router upgrade thresholds 1, 2, and 4 for thresholds gaps 10, 5, and 1, respectively. It is worth noting that the retransmission ratio may not strongly correlate with total transmitted frames in the network, which can be highlighted in the case of router upgrade threshold 3, where there was a slightly higher overall number of transmitted frames. However, the retransmission ratio was lower in comparison with neighboring router upgrade threshold settings. The lack of strong correlation between frame retransmissions and total transmitted frames may be caused by differences in communication between routers in the network.

It was observed that, even though the router upgrade and downgrade thresholds are set to specific values on each REED node, the actual number of routers in the network might not correspond to those values. Router degradation to REED is only possible if no child node is attached to it. Figure 8 shows the actual number of routers in the network (routers + border router), which was a minimum of 4 to 5, despite the lower router downgrade threshold. Different upgrade and downgrade threshold hysteresis is supposed to minimize role changes. However, in our topology, it does not have a significant effect. This can be seen in Figure 9, where cumulative role change is summed over repeated 20 runs for each router upgrade threshold.

From the results achieved in this scenario, it can be concluded that different gaps between router upgrade and router downgrade thresholds have a negligible impact on the network performance. While it seems that the router upgrade threshold can affect the

performance significantly, a low router upgrade threshold may lead to a higher number of router changes in the network. Moreover, it may also increase the frame retransmission ratio in some cases.

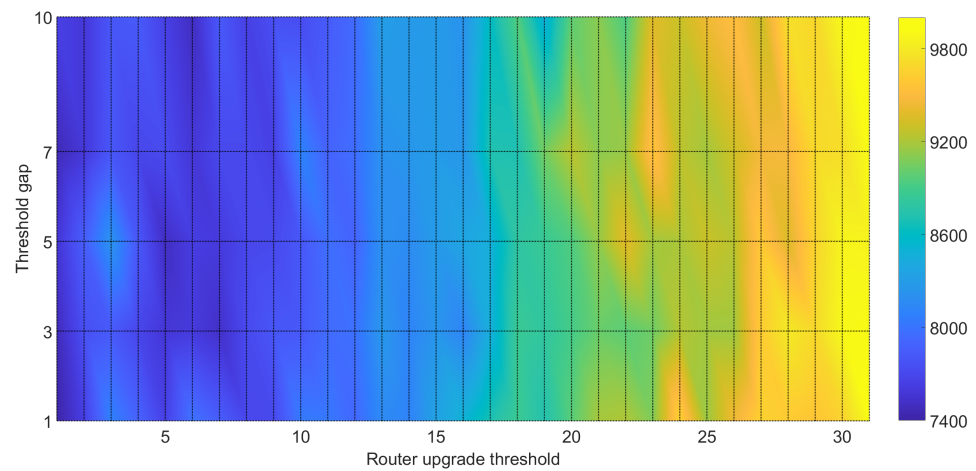


**Figure 6.** Mean number of Tx frames for different thresholds and threshold gaps over 20 simulation runs.

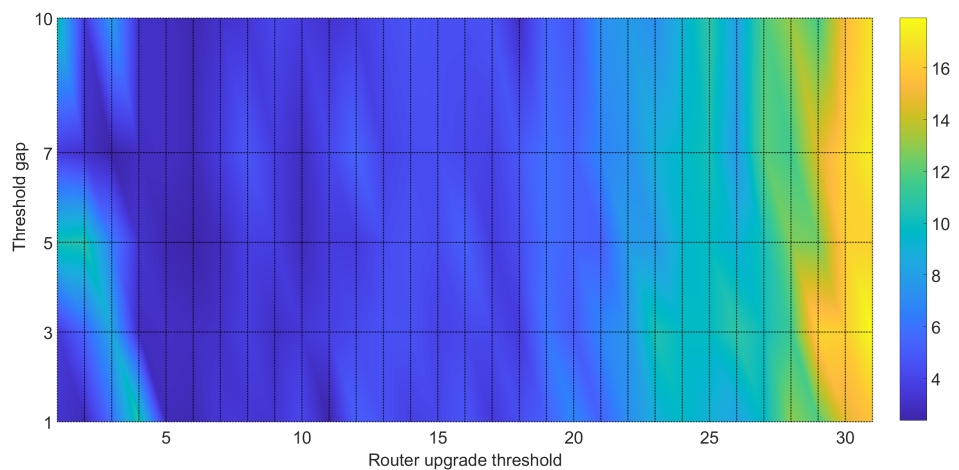


**Figure 7.** Tx retry ratio [%] for threshold gap comparison, based on 20 simulation runs.

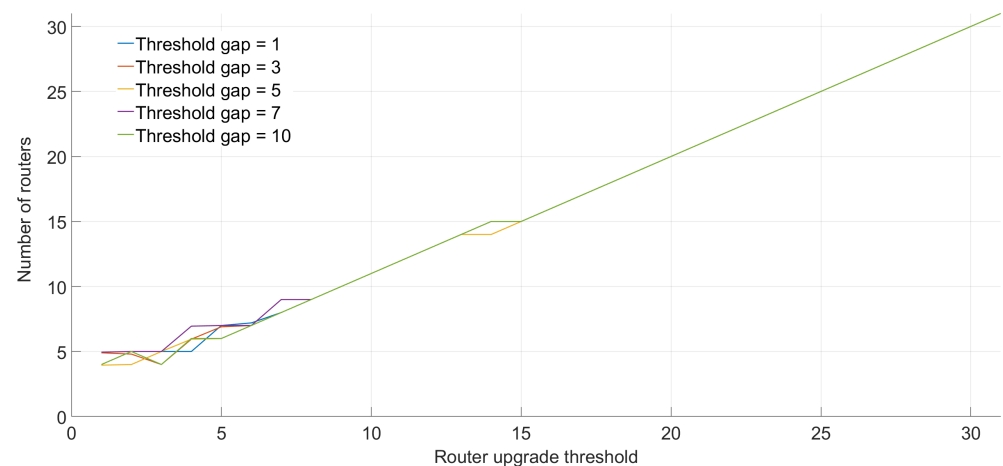


**Figure 8.** Mean number of routers in network for threshold gap comparison over 20 simulation runs.

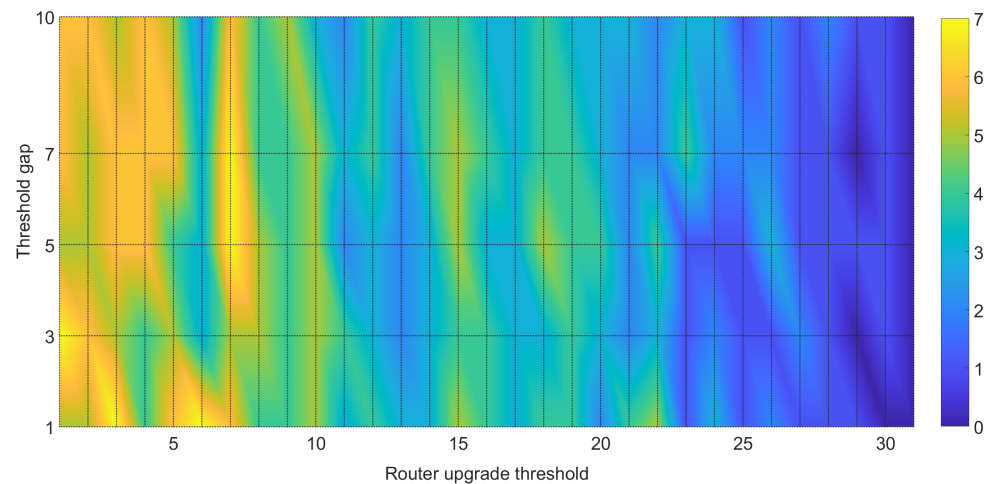


**Figure 9.** Cumulative role changes for different thresholds and threshold gaps over 20 simulation runs.

On the other hand, a higher router upgrade threshold leads to lower changes in the roles of REEDs. However, it was observed that it also results in a higher number of transmitted frames as well as a higher frame retransmission ratio, thus making the data transfers in the network less efficient.

#### 4.2. Scenario 2: Impact of the Network Topology

In the second scenario, we evaluated the performance of the OpenThread network in three grid topologies with identical node counts but increasing spacing of 10 m, 20 m, and 30 m between REEDs. Additionally, one heterogeneous topology with randomly distributed 30 REEDs in an area 50 m × 60 m was added. In all tested topologies, CoAP messages with 50 B payload were sent each second from a randomly selected node to the BR. The upgrade/downgrade threshold gap was fixed at 3 while we swept the router upgrade threshold. Our analysis focuses on how node spacing and layout influenced the following:

- CoAP message latency.
- Actual number of routers for different router upgrade thresholds.
- Ratio of MAC frame retransmission to total transmitted MAC frames.
- Network transmission time for different router upgrade thresholds.

In Figure 10 the latency of CoAP messages for router upgrade thresholds in different topologies is shown. As expected, smaller topologies have lower latency of CoAP messages. This is due to the fewer multi-hop message transmissions required to reach the BR. The router upgrade threshold and actual number of routers in the network are shown in Figure 11. The results indicate that the actual number of routers in the network does not have a significant impact on message latency.

Despite the configured router upgrade and downgrade thresholds set for each run in topologies, the actual number of routers can differ. In Figure 11, the average number of routers in each topology is shown, based on data collected from 20 runs for each router upgrade threshold. Depending on the topology size, the actual number of routers was higher than the actual router downgrade threshold, especially in the topology with 30 m node spacing. In this case, the lowest average number of active routers was 11.95 when the router upgrade and downgrade thresholds were set to 3 and 6, respectively.

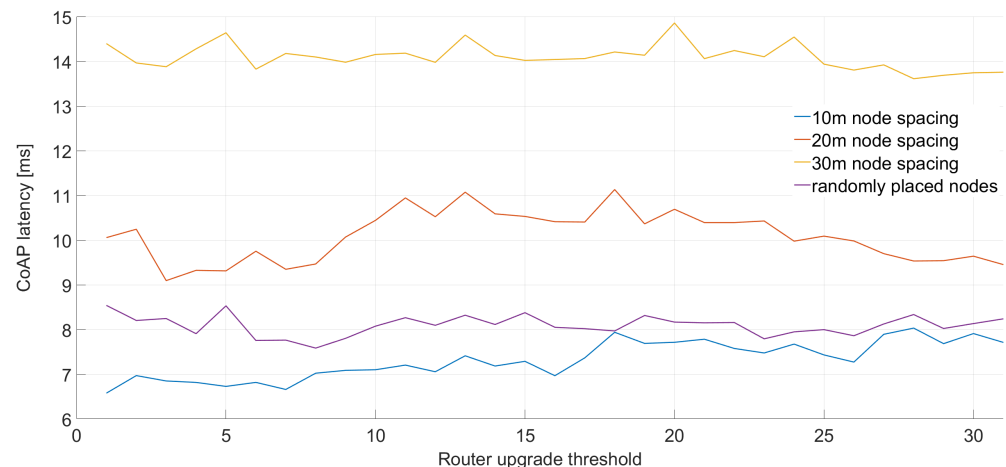


**Figure 10.** Mean CoAP latency in different topologies comparison based on 20 simulation runs.

The reason for this is that active routers with a single child node attached cannot demote themselves from the router role. Similar behavior can be observed in a topology with 20 m node spacing, in which the lowest average number of active routers was 7. This value was again achieved with the router upgrade and downgrade thresholds of 3 and 6. In the topology with randomly distributed nodes minimal number of routers was achieved at router upgrade thresholds set to 3 and 6, where the actual number of active routers was 6. In comparison with grid topology and 10 m node spacing, which covers an area of similar size, grid topology achieved a smaller number of active routers within the network. Notably, the minimum number of active routers observed was 4, occurring at a router upgrade threshold set to 3. This difference may be related to the distribution of nodes in the area, where grid topology provides better coverage with fewer router nodes.



**Figure 11.** Mean number of routers in different topologies based on 20 simulation runs.

A lower number of routers in the network results in decreased network traffic due to network management overhead and periodic advertisements from routers. On the other hand, it may not be sufficient to cover the whole area and may result in increased message retransmission. Figure 12 shows the ratio between retransmitted frames and overall frame transmissions. In grid topologies with 10 m and 30 m spacing, it is possible to observe increased retransmission ratios in lower upgrade threshold ratios.

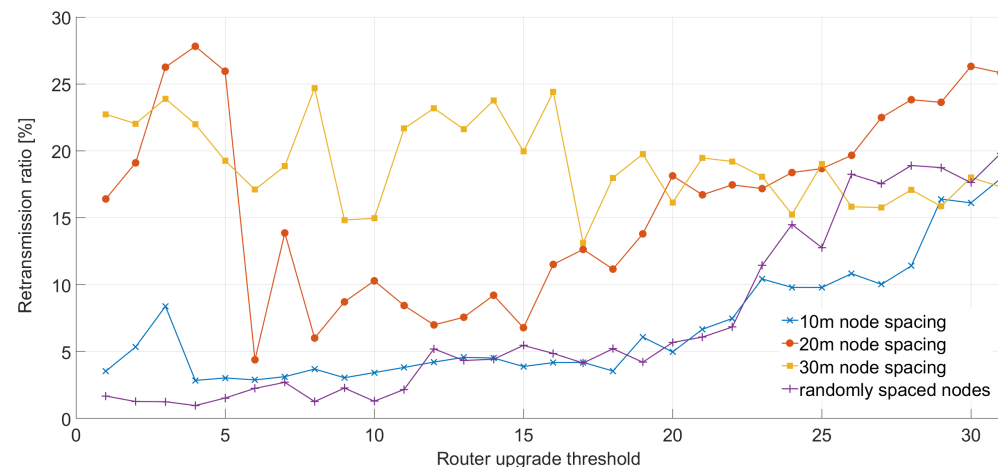


**Figure 12.** Ratio between MAC frame retries and total transmitted MAC frames based on 20 simulation runs.

In Figure 13, the transmission times of tested topologies across the router upgrade threshold sweep are shown. The figure shows the mean value of transmission times summed over all nodes for each simulation run. The topologies with a smaller footprint, namely, 10 m node spacing and randomly placed nodes, show increased overall transmission times with rising upgrade thresholds, specifically from 26.78 s to 41.67 s and 29.28 s to 43.78 s in those topologies, respectively.

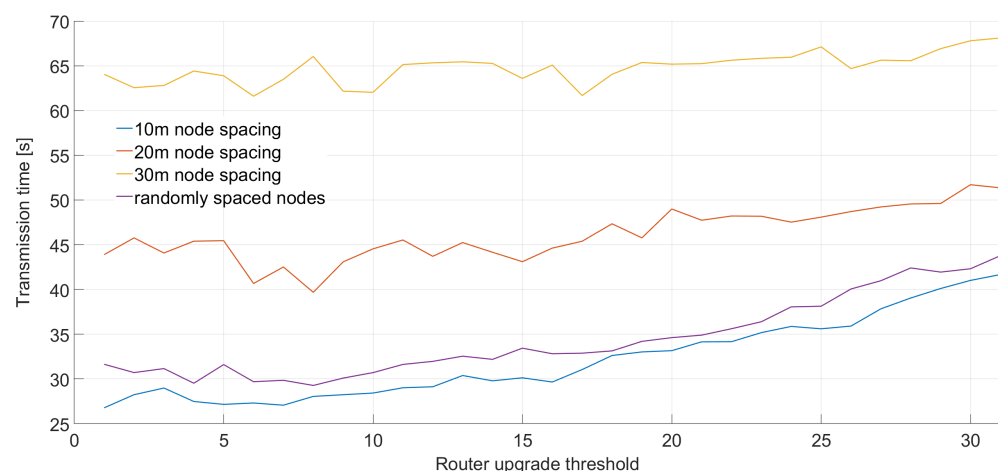


**Figure 13.** Mean transmission time in different network topologies based on 20 simulation runs.

Topologies with larger spacing indicate similar increased transmission times with higher router upgrade thresholds; however, the impact is not that significant. In the case of a topology with a 20 m node spacing, the lowest transmission time is 39.69 s with a router upgrade threshold of 8, and the highest transmission time is 51.71 s with a threshold of 30. The topology with 30 m node spacing achieved the lowest transmission time of 61.6 s and a maximum transmission time of 68.1 s for a router upgrade threshold of 6 and 31, respectively. Please note that the higher retransmission ratios shown in Figure 12 generally translate into longer transmission times, especially in 10 m and 20 m node spacing and randomly distributed nodes. In the case of the 30 m spacing, the retransmission ratios stay high despite the actual number of routers in the network.

#### 4.3. Scenario 3: Impact of the Payload Size

To observe any performance changes, in this simulation scenario, CoAP messages with different payload sizes were sent from REEDs to the BR in a grid topology with 10 m node spacing, with the gap between router upgrade and downgrade thresholds set to 3. Each CoAP message was sent randomly from a different REED every second to BR. The constant payload sizes were defined as follows:

- No CoAP message was sent; no additional traffic.
- CoAP messages with 10 B payload.
- CoAP messages with 50 B payload.
- CoAP messages with 100 B payload.

Please note that transmission times of each CoAP message may vary depending on multi-hop transmission and packet fragmentation as described in Section Testing Scenarios. The detailed sizes of MAC frames and their transmission times are shown in Table 2. Moreover, from the table, it is clear that in the case of CoAP messages with a 100 B payload, frame fragmentation will occur. Thus, it can be expected that the number of transmitted frames will increase in this case.

Figure 14 shows the relationship between the router upgrade threshold and the total number of frames transmitted in the case when no CoAP messages were sent to the Border router. Therefore, only Thread management and advertisement traffic were transferred in the network. The boxes in the figure show the upper and lower quartiles and the line in the box shows the median values, while the whiskers below and above the box show the minimum and maximum values, respectively. Moreover, outlier values are shown as circles, and a red x represents the mean value.

From the figure, it can be seen that with an increase in the router upgrade threshold parameter, the total number of transmitted frames increases. This can be expected due to the increased number of routers, which generate more traffic than REED devices.

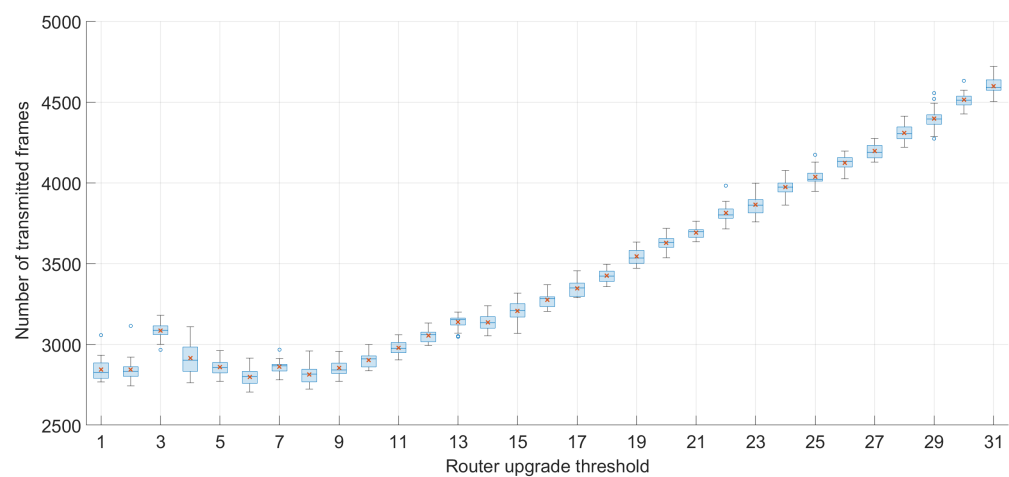


**Figure 14.** Number of transmitted frames without CoAP transmissions based on 20 simulations.

However, the global minimum of the transmitted frames is not observed at the lowest router upgrade threshold setting (threshold = 1), but rather at threshold = 6, with a mean of 2798 frames transmitted. An increase in the total number of transmitted frames translates to longer cumulative transmission times across the entire set of nodes in every payload profile. It should be noted that the response to the router threshold is non-linear, with a minimum at 6 for the router upgrade threshold. We assume this is due to the fact that a lower value of the router upgrade threshold results in sub-optimal coverage and a higher number of retransmissions due to a higher frame error rate.

In Figure 15, the mean transmission time of all nodes is shown for each payload profile. An increase in cumulative transmission time between different payload sizes is due to longer transmission times of messages sent across the network, and in the case of a CoAP message with a 100 B payload, the packet size is larger than the MTU of the 802.15.4 frame, so LoWPAN packet fragmentation is used to split the message into two frames (see Table 2).

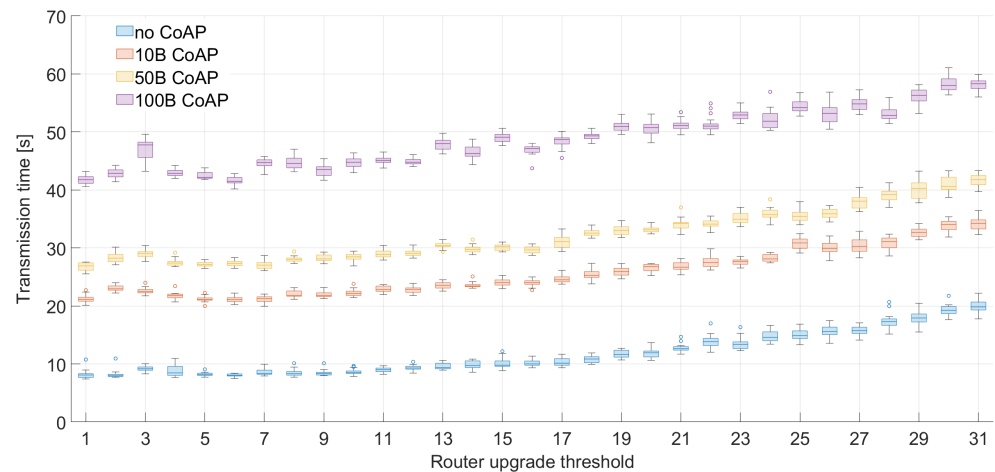


**Figure 15.** Transmission time of all nodes in the network for 20 simulations.

A comparison of minimum and maximum transmission times for each payload profile is shown Table 3, revealing that the time difference ( $\Delta$  Tx time) extends from 11.86 s in the case with no CoAP messages to 16.72 s in the case of a CoAP message with 100 B payload as the router upgrade threshold (i.e., the required number of upgraded routers) increases. The global minimum in transmission time was achieved with a router upgrade threshold of 6 for cases of 10 B, 100 B, and no CoAP message. While the global minimum for 50 B CoAP message payload was achieved with router upgrade threshold 1. The concurrent rise in both bounds of transmission time implies heavier overall network load at higher thresholds, elevating the risk of unnecessary congestion. Moreover, it may result in higher energy consumption for the devices.

**Table 3.** Difference of mean transmission times.

| CoAP Payload | Min Tx Time [s] | Max Tx Time [s] | $\Delta$ Tx Time [s] |
|--------------|-----------------|-----------------|----------------------|
| no CoAP      | 8.06            | 19.92           | 11.86                |
| 10 B CoAP    | 21.14           | 34.11           | 12.97                |
| 50 B CoAP    | 26.78           | 41.67           | 14.89                |
| 100 B CoAP   | 41.58           | 58.29           | 16.72                |

In Figure 16, the mean of the total transmitted frames for each scenario is shown. A similar trend of increased total transmitted frames with an increase in the router upgrade threshold parameter is observed in each payload profile. The higher number of transmitted frames in the case of sending CoAP messages with 100 B payloads is caused by the fragmentation of the message. Moreover, it should be noted that the number of transmitted frames for both 10 B and 50 B payloads is almost the same. Therefore, it can be concluded that the difference in the transmission time depicted in Figure 15 is mainly caused by the different lengths of the transmitted frames, since longer frames require longer transmission intervals.

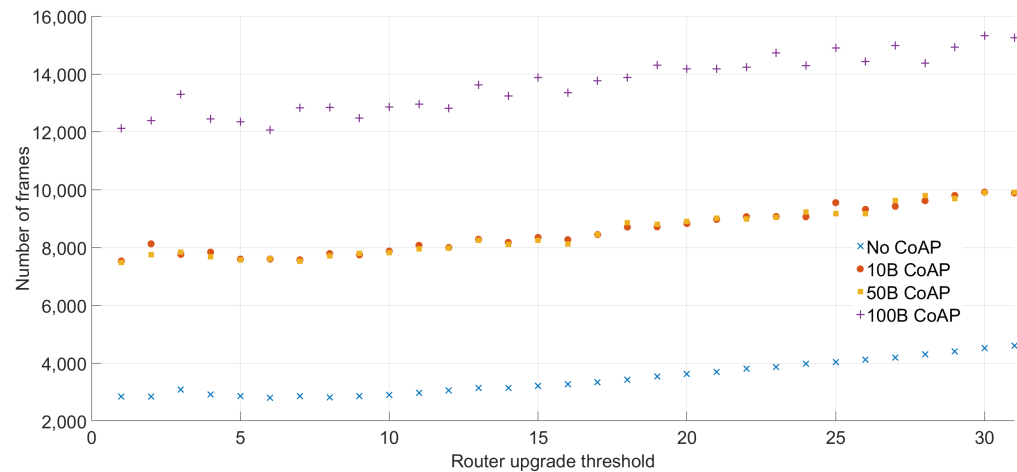


**Figure 16.** Mean number of transmitted frames in all scenarios based on 20 simulations.

The increased total number of transmitted frames for higher router upgrade thresholds is caused by the increased number of frame transmission retries, as well as by additional advertisement messages from routers. In Figure 17, the ratio between the number of message transmission retries and the total number of transmitted frames is shown. In the case when the router upgrade threshold was set to 6, it is possible to see the lowest transmission retry ratio. Therefore, it can be concluded that this upgrade threshold provides the most efficient settings for the given network topology. This results in reduced transmission retries as well as total transmission time. In the case where no CoAP traffic was generated, we observed the worst ratios across the entire router upgrade threshold sweep. Although all payload profiles show a similar trend with an increase in router upgrade thresholds.

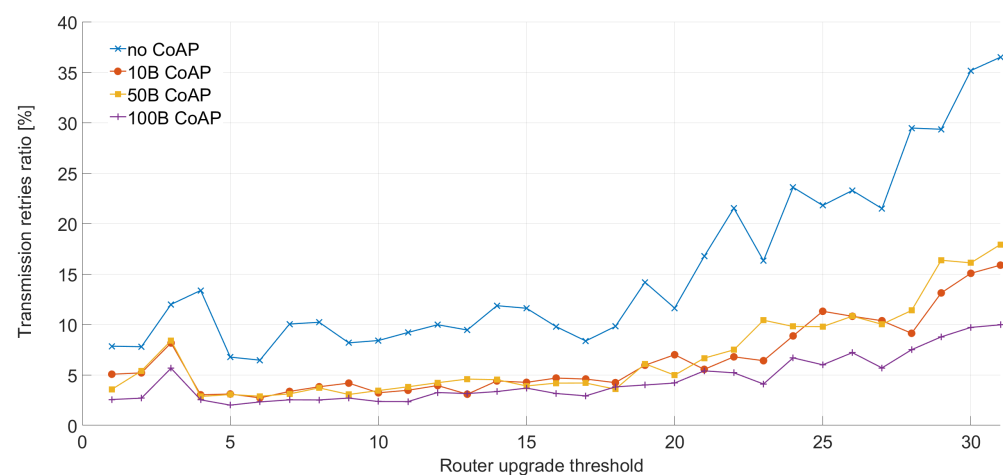


**Figure 17.** Ratio of transmission retries to total transmitted frames based on 20 simulation runs.

The likely cause of this could be the same time of periodic transmission of messages that maintain links between neighboring routers, which can be seen on Figure 18. From the figure, the increased number of dropped frames at the receiver due to FCS error can be seen. The results show a similar number of dropped frames due to FCS errors for all four payload profiles, despite the different numbers of total transmitted frames. Thus, it can be concluded that increased periodic traffic causes worse SIR at the node receivers, eventually leading to more frame drops.

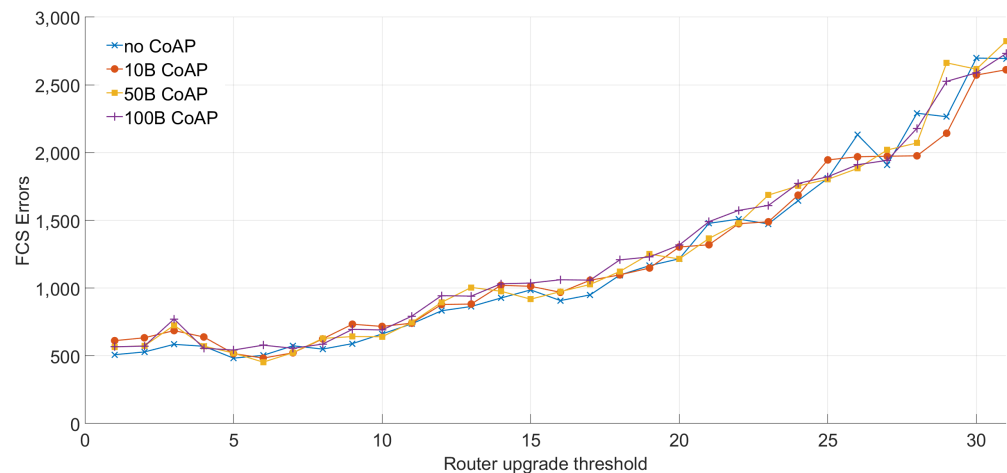


**Figure 18.** Mean number of frames dropped due to FCS error over 20 simulation runs.

Although each experimental scenario specified a router upgrade threshold in the range of 1–31, this parameter should not be interpreted as the actual number of routers deployed in the simulated network. The Thread network maintains CDS via a distributed algorithm and tries to minimize the number of detached nodes and network partitions.

In Table 4, the KPI parameters achieved for the default router upgrade and downgrade thresholds are compared with the KPIs achieved for thresholds selected based on simulation results. The KPIs are shown for two different payloads, i.e., no CoAP messages and 50 B CoAP messages.

**Table 4.** Comparison Thread default router upgrade and downgrade thresholds (16/23) with thresholds selected based on simulation results (6/9).

| KPI            | No CoAP |       | 50 B CoAP |        |
|----------------|---------|-------|-----------|--------|
|                | 16/23   | 6/9   | 16/23     | 6/9    |
| Tx retries [%] | 11.67   | 6.46  | 4.11      | 2.89   |
| FCS errors     | 972.55  | 503.7 | 945.15    | 453.15 |
| Tx time [s]    | 10.44   | 8.06  | 30.19     | 27.31  |

The results shown in the table show that Tx retries were reduced by 44% when traffic did not include CoAP payload and by 29% when 50 B CoAP messages were transmitted. Moreover, it is also possible to see that changing the thresholds had a positive impact on the number of FCS errors, reducing them by 48% and 52%, respectively. This represents quite a significant improvement in the performance of the Thread network. Moreover, it was also possible to reduce the transmission time in the simulation run by more than 2 s in each case. Therefore, it can be concluded that changing the router upgrade and downgrade thresholds can help to reduce Tx time and thus reduce energy consumption of the nodes in the network.

In summary, the achieved results at the tested topology indicate that optimal performance, in terms of mean transmission time, is achieved when the router upgrade threshold is set between five and seven across all four payload profiles. Further reduction in upgrade/downgrade thresholds leads to increased frame loss due to insufficient coverage. Conversely, higher router upgrade/downgrade thresholds introduce higher interference caused by excessive transmission from neighboring routers, thereby increasing overall air time and reducing network efficiency.

Thus, the thresholds could be modified if the efficient transmission of data in the network is required, for example, due to energy constraints. Since there is currently no option for the automatic setting of threshold, we suggest running a simulation with different router upgrade thresholds for the deployed topology of a Thread network and modifying the thresholds based on the achieved results. Development of an algorithm for the automated selection of router upgrade and downgrade thresholds is envisioned as part of the future work.

## 5. Conclusions and Future Work

This work set out to quantify how the two Thread parameters that govern router formation, i.e., the router upgrade threshold and the router downgrade threshold applied to REED (Router-Eligible End Device) nodes, influence overall medium utilization. The network performance was tested using controlled test topologies under three scenarios while systematically sweeping these thresholds.

Across all three scenarios, we found that Thread performance is related to router density. In the first scenario, with 30 REEDs distributed in a grid with 10 m spacing, we observed that very low router upgrade thresholds (2–5) caused large swings in total frames and more than 10% retransmission ratios for some threshold gaps, but hysteresis had little impact on role stability. The network still kept about 4–5 routers because routers with children could not demote to maintain CDS.

In the scenario, different topologies were examined, and the same nominal settings produced very different router counts in the network, starting from 4 routers in the case of 10 m spaced REEDs to 12 active routers for the sparse 30 m spaced REEDs. The regular 10 m grid covered the area with fewer routers compared to the random layout, confirming that structured placement can reduce router count without harming connectivity. However, pushing the router count too low increased retransmissions and transmission time, while sparse 30 m spacing suffered high retransmissions regardless of the thresholds due to limited link quality.

Across every scenario, a larger active-router population translated into a higher volume of MAC-layer frames on the air. Two phenomena driving this trend are as follows:

- Each additional router injects extra management traffic (advertisements, MLE, and address-solicit/notify exchanges).
- Denser router deployments amplify intra-network interference, increasing the retransmission rate.

Together, these factors produced a measurable rise in total frame count and, therefore, medium utilization. The findings show that the default threshold pair is not universally optimal. In the topologies presented in this paper, the default router upgrade threshold is redundant.

It was shown that default threshold settings could lead to increased air time and more data retransmissions due to collisions. Conversely, overly restrictive thresholds risk under-provisioning the routing backbone and harming network performance. Maintaining a moderate excess of routers is sufficient for reliability, while an excessive number unnecessarily burdens the radio channel.

Finally, the study highlights two avenues for future work. First, dynamic algorithms that adapt the thresholds in real time will be developed. These algorithms could better track changing traffic and could be used to optimize network traffic. Second, the results will be further validated in larger and more heterogeneous environments. These could include mixed-vendor hardware and real-world radio conditions. Nonetheless, the present simulations provide evidence that thoughtful threshold selection is a simple yet effective lever for managing medium utilization in Thread networks and can be used as a method to

optimize the thread network. By changing the router upgrade and downgrade thresholds, it is possible to reduce errors in transmissions and reduce the transmission time of nodes in the network, resulting in more energy-efficient data transfers.

**Author Contributions:** Conceptualization, X.Z. and J.M.; methodology, X.Z.; software, X.Z.; validation, X.Z., J.M., and P.B.; formal analysis, X.Z.; investigation, X.Z.; resources, J.M. and P.B.; data curation, X.Z.; writing—original draft preparation, X.Z.; writing—review and editing, J.M. and P.B.; visualization, X.Z.; supervision, J.M. and P.B.; project administration, J.M.; funding acquisition, P.B. and J.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work has been supported by the Slovak VEGA grant agency, Project No. 1/0580/25 “Research of Effective Localisation and Communication Solutions in Heterogenous Wireless Networks”.

**Data Availability Statement:** The original data presented in the study are openly available in a GitHub repository available at <https://github.com/xaver-zak/Impact-of-router-count-in-OpenThread> (accessed on 20 November 2025)

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Thread Group. Thread Benefits. Available online: <https://www.threadgroup.org/What-is-Thread/Thread-Benefits> (accessed on 10 November 2025).
2. OpenThread. Node Roles and Types | OpenThread. Available online: <https://openthread.io/guides/thread-primer/node-roles-and-types> (accessed on 10 November 2025).
3. OpenThread. Definitions for MLE Functionality Required by the Thread Child, Router, and Leader Roles. Available online: <https://github.com/openthread/openthread/blob/main/src/core/thread/mle.hpp> (accessed on 10 November 2025).
4. Görgülü, H.B.; Sari, K.; Çiraci, E.; Taş, M.O. Energy Efficient and Scalable Wireless Sensor Networks for Industrial Areas Using OpenThread. In Proceedings of the 2024 Innovations in Intelligent Systems and Applications Conference (ASYU), Ankara, Türkiye, 16–18 October 2024; pp. 1–5. [CrossRef]
5. Checco, A.; Fabiani, V.; Palmisano, M.; Polese, D. Network Connectivity and Sensor Response of an OpenThread WSN Platform for Crop Monitoring. *IEEE Trans. AgriFood Electron.* **2024**, *2*, 218–225. [CrossRef]
6. Sittampalam, G.; Namal, S.; Ragel, R. Two-Factor Commissioning for Thread Protocol. In Proceedings of the 2023 3rd International Conference on Advanced Research in Computing (ICARC), Belihuloya, Sri Lanka, 23–24 February 2023; pp. 190–195. [CrossRef]
7. Akestoridis, D.G.; Sekar, V.; Tague, P. On the Security of Thread Networks: Experimentation with OpenThread-Enabled Devices. In Proceedings of the WiSec '22: 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks, New York, NY, USA, 16–19 May 2022; pp. 233–244. [CrossRef]
8. Mihaeljans, M.; Skrastins, A. Openthread Network Density Evaluation: Quantitative Analysis. In Proceedings of the 2023 Symposium on Internet of Things (SIoT), São Paulo City, Brazil, 25–27 October 2023; pp. 1–5. [CrossRef]
9. Seehamat, P.; Khunboa, C. The Performance Study on CoAP Protocol over an OpenThread Network. In Proceedings of the 2025 13th International Electrical Engineering Congress (iEECON), Hua Hin, Thailand, 5–7 March 2025; pp. 1–5. [CrossRef]
10. Banaie, F.; Misic, J.; Misic, V.B.; Yaghmaee Moghaddam, M.H.; Hosseini Seno, S.A. Performance Analysis of Multithreaded IoT Gateway. *IEEE Internet Things J.* **2019**, *6*, 3143–3155. [CrossRef]
11. Khattak, S.B.A.; Nasralla, M.M.; Farman, H.; Choudhury, N. Performance Evaluation of an IEEE 802.15.4-Based Thread Network for Efficient Internet of Things Communications in Smart Cities. *Appl. Sci.* **2023**, *13*, 7745. [CrossRef]
12. Johnson, B.; Lazar, A.; Woodward, C. *Thread 1.2 Base Features*; Thread Group, Inc.: San Ramon, CA, USA, 2019.
13. *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)*; IEEE Standard for Low-Rate Wireless Networks. IEEE: Piscataway, NJ, USA, 2016; pp. 1–709. [CrossRef]
14. Silicon Laboratories. *OpenThread Developer's Guide v2.5.0*; Chapter Synchronized Sleepy End Devices (SSEDs); Silicon Laboratories: Austin, TX, USA, 2024.
15. OpenThread. Network Discovery and Formation. Available online: <https://openthread.io/guides/thread-primer/network-discovery> (accessed on 11 November 2025).
16. OpenThread. Router Selection. Available online: <https://openthread.io/guides/thread-primer/router-selection> (accessed on 12 November 2025).
17. OpenThread. Simulation. Available online: <https://openthread.io/platforms/simulation> (accessed on 13 November 2025).
18. OpenThread. Co-Processor Designs. Available online: <https://openthread.io/platforms/co-processor> (accessed on 10 November 2025).

19. OpenThread. OpenThread Network Simulator, a Discrete Event Simulator and Visualization Tool for Thread Networks. Available online: <https://github.com/openthread/ot-ns> (accessed on 13 November 2025).
20. 3GPP. *Study on Channel Model for Frequencies from 0.5 to 100 GHz (3GPP TR 38.901 Version 17.0.0 Release 17)*; TR 138 901—V17.0.0—5G; 3GPP: Sophia Antipolis, France, 2022.
21. *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*; IEEE Standard for Information Technology—Local and Metropolitan Area Networks—Specific Requirements—Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs). IEEE: Piscataway, NJ, USA, 2006; pp. 1–320. [CrossRef]
22. NS3. ns-3 LrWpanErrorModel Class Reference. Available online: [https://www.nsnam.org/docs/release/3.29/doxygen/classns3\\_1\\_1\\_lr\\_wpan\\_error\\_model.html](https://www.nsnam.org/docs/release/3.29/doxygen/classns3_1_1_lr_wpan_error_model.html) (accessed on 14 November 2025).
23. NS3. ns-3 Low-Rate Wireless Personal Area Network (LR-WPAN) Model Library. Available online: <https://www.nsnam.org/docs/release/3.25/models/singlehtml/index.html#document-lr-wpan> (accessed on 12 November 2025).
24. Shelby, Z.; Hartke, K.; Bormann, C. The Constrained Application Protocol (CoAP). RFC 7252, 2014. Available online: <https://www.rfc-editor.org/info/rfc7252> (accessed on 14 November 2025).
25. Thubert, P.; Hui, J. Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks. RFC 6282, 2011. Available online: <https://www.rfc-editor.org/info/rfc6282> (accessed on 14 November 2025).
26. Deering, D.S.E.; Hinden, B. Internet Protocol, Version 6 (IPv6) Specification. RFC 8200, 2017. Available online: <https://www.rfc-editor.org/info/rfc8200> (accessed on 13 November 2025).

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.