Collision Minimization Beacon Scheduling Scheme using RPL in Dense TSCH-based IoT Environment

Jae-Young Kim
Pusan National University
Sang-Hwa Chung (shchung@pusan.ac.kr)
Pusan National University https://orcid.org/0000-0003-1329-1188
Gihwan Kim
Pusan National University
Minjae Kim
Pusan National University

Research

Keywords: IEEE 802.15.4e, TSCH, Industrial WSN, Beacon collision, Network formation, Fast joining

Posted Date: July 17th, 2020

DOI: https://doi.org/10.21203/rs.3.rs-38962/v1

Collision Minimization Beacon Scheduling Scheme using RPL in Dense TSCH-based IoT Environment

Jae-Young Kim, Sang-Hwa Chung*, Gihwan Kim, Minjae Kim

Department of Electrical and Computer Engineering, Pusan National University, Busan, Republic of Korea

Email: ttled45@gmail.com, shchung@pusan.ac.kr*, gihwan@pusan.ac.kr, minjae@pusan.ac.kr

Abstract

IEEE 802.15.4e Time Slotted Channel Hopping (TSCH) operates on a time-division scheme and uses 16 separate channels for each communication to ensure high reliability even in an industrial environment where many devices are concentrated. The process of participating in the network is indispensable because the entire network operates at the same time. However, there is a problem with TSCH's channel change technique, which results in longer network participation time for new nodes wanting to participate in the network. Previous research has randomly chosen channels for beaconing to reduce network formation time. However, this approach is effective for networks with fewer nodes, but conflicts in beacon messages in networks with approximately 20 or more networks result in network congestion, which increases network participation time. To solve this problem, this paper proposes a collision minimization scheduling technique. This strategy assigns a separate time zone to each node to exclude the possibility of beacon message collisions. This algorithm allocates timeslot based on RANK of RPL. By explicitly predicting the longest network joining time of the network participating nodes, the average network joining time was measured to be about 25 seconds to 27 seconds faster compared to the previous research.

Keywords

IEEE 802.15.4e; TSCH; Industrial WSN; Beacon collision; Network formation; Fast joining

Introduction

Today, the Internet of Things (IoT) has become common. Needs for the IoT have increased in various fields such as home networks where simple connections between home electronics are established or industrial networks where facility data generated by sensors are transmitted and received. Especially, a lot of large-scale dense IoT networks which numbers of sensors form clusters are emerging from small sensor networks in the past. Smart factory is a use case of the large dense networks. Sensors are attached to facilities and help to determine next action by uploading information. In addition, facility managers can monitor status of the facility and products being manufactured in real-time by connecting to the factory network. To meet these demands, there is growing interest in low-power wireless sensor networks that can reduce wiring costs for connecting the sensors and can be installed in a mobility device such as a robot.

IEEE 802.15.4e[1] TSCH MAC and RPL routing technology are typical technologies for low-power wireless sensor networks. Time Slotted Channel Hopping (TSCH) MAC is a technology for low power sensor networks. It has multi-channel and channel hopping, which is exceptionally reliable in congested situations and is well received in industrial environments that require stringent standards. IPv6 Routing Protocol for low-power Lossy Networks (RPL) [2] is created for low-power sensor networks. Sensors use control messages to maintain the network. It is specialized in low power IoT environment because it can flexibly generate path with low overhead.

There have been many attempts to configure the low-power IoT wireless network through TSCH and RPL to be used in an industrial environment. Since the TSCH network exchanges data after the entire network is synchronized, the TSCH performs not well in a situation where frequent re-joining occurs. However, most of the studies in TSCH focus on data exchange after synchronization.

In this paper, we focus on the formation of the TSCH networks. It is a process to inform to outside about the network periodically and a starting point for new nodes to join the network. Nodes in the network broadcast Enhanced Beacon (EB) to neighbor nodes. Nodes not joined yet select one of the 16 channels according to the standard and turn on the radio and wait for the EB reception. The EB contains information about the network. So, nodes can join the network based on the information. However, since the radio must be turned on before receiving the EB, energy consumption increases in proportion to waiting time of the EB. It can be fatal problem for devices with small battery capacity. That is, differences in the network join time can affect performance in terms of network lifetime.

Various schemes have been proposed to supplement the basic strategy of the TSCH standard for network formation. Most of these take ramdom-based strategies that can be useful when the number of nodes is below the TSCH channel number within the communication range. However, in large wireless industry sensor network environments where many devices are concentrated, EB transmission and reception collisions between devices can occur frequently, resulting in long EB reception delays.

Suppose an industrial environment is congested with more than 100 nodes. Each node will try to access the network without any interferences. Because, there can be many different network engagement processes and differences in network formation play a sensitive role in data collection and device performance, a strategy for rapid network formation is essential.

Therefore, in a crowded environment where many nodes are connected, without a deterministic strategy for network formation, energy is wasted for a long time before joining the network. This problem can be a weakness for industrial networks that must provide high reliability.

We propose a method to join the large-scale wireless industrial network environment with minimal EB collision. Each node allocates a cell for EB transmission based on RPL rank. It reduces EB reception delay by eliminating the EB collision which occurs during the network formation.

The main contributions of this paper are as follow:

- We solved the uncertainty of the maximum network participation time of the random-based technique.
- We proposed a technique to minimize network congestion by minimizing data collision during network formation.
- EB scheduling was attempted in the TX slot rather than the existing ADV slot, allowing the EB to broadcast throughout the slot frame.
- We implemented our scheme using the OpenMote-CC2538 [10] ported to the OpenWSN [11] project based on 802.15.4e TSCH.

Thus far, many papers adapted a random-based scheme which broadcast EBs on a random in small networks. However, with an emphasis on the need for wireless networks in industrial environments, there are increasing instances of deploying wireless networks in crowded or mobile environments. In such a large-scale wireless network, communication-enabled devices are installed in a concentrated manner, and data exchanges between devices are frequently conducted. In such an environment, a randomly running strategy presents a risk in that applications using the network may fall into an infinite waiting state owing to the unpredictability. In addition, there is no way to prevent collisions under a crowded situation when EBs are sent on a randomly selected channel. In fact, the 2.4 GHz ISM band, which is used in the IEEE802.15.4e standard, is used by many communication protocols, including ZigBee and Wi-Fi. Therefore, the collisions between EBs in a state in which congestion occurs to a certain extent further deepens this problem..

Background

This paper describes a scheme for shortening the synchronization time of the TSCH network based on the information obtained from the RPL. In this section, we describe IEEE802.15.4e TSCH, TSCH network formation, how to build DODAG of RPL protocol and DIO and DAO message exchange method to maintain it.

IEEE 802.15.4e TSCH

The 802.15.4e standard is a revision of the previous 802.15.4 standard to use in an industrial environment. This standard supports three MAC behavior modes. Each mode is made for a specific application and has its own characteristics, sharing the basic concept. Among them, we focus on the Time Slotted Channel Hopping (TSCH) mode. TSCH is intended for industrial environment automation and is a time division access technology supporting multi-channel and multi-hop communication. The characteristics of the TSCH are that it is divided into 16 channels, and the channel is changed by changing the frequency periodically. First, since the time division multiple access is used, it is possible to prevent collision between competing nodes. Thus, it can increase the potential throughput. In addition, since 16 channels are used, data can be exchanged between many nodes at the same time, thereby increasing the total amount of data in the network. In addition, the channel change technique mitigates congestion and multipath fading due to jamming to ensure communication reliability. In the TSCH, all the nodes participating in the network are synchronized to the continuously repeated slotframe. slotframe consists of N_{slot} timeslot of a certain length. Each timeslot has enough length for each node to send and receive data packet and acknowledgments. If communication is not properly established due to a transmission problem, the next slotframe will be retransmitted. Figure. 1 shows a slotframe consisting of four timeslots.

Another feature of the TSCH explicitly divides the channel into 16 channels. Also, it changes periodically and communicates. Basically, all 16 channels are used, or several channels are selected depending on the implementation specification. The channel N_{ch} used for changing the channel is composed of 16 or less. The physical frequency f used for actual communication in the channel change is determined by the channel offset set in the timeslot. The following equation 1 is for determining the actual frequency.

$$\text{frequency} = \text{F}[(\text{ASN} + \text{channelOffset})\ mod\ N_c] \qquad (1)$$

In the equation 1, the Absolute Slot Number (ASN) is the total number of timeslots that have flowed from the beginning of the network to the present. This value is shared by all the nodes and synchronized to the network. F is implemented as a lookup table, which contains information about the channel sequence.

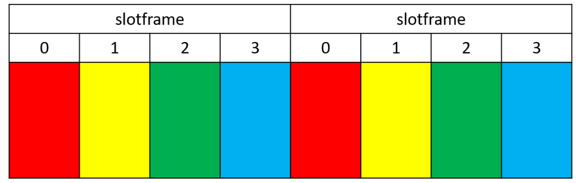

Figure 1. The TSCH slotframe structure.

In the TSCH, each link communicates in one slotOffset and channeloffset, and is called as TSCH cell on {slotOffset, channelOffset} tuple. Because the ASN is changing, physical channel f determined by equation (1) is continuously changed. Because multiple channels are used, each node in the network can communicate multiple times in the same slotOffset. It also alters the channel at every timeslot, thus mitigating the negative effects of external congestion.

The cell of the TSCH network consists of dedicated cell and shared cell. In the case of dedicated cells, only one link can be communicated across the network. In the case of the shared cell, various links send / receive various messages competitively.

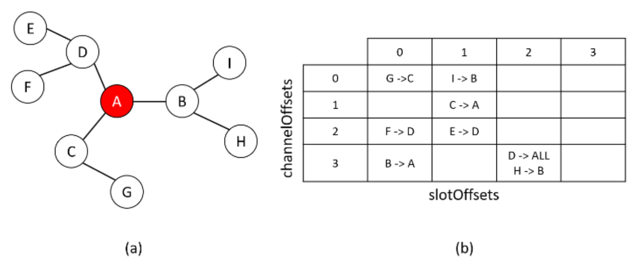

Figure 2. (a) TSCH network topology and (b) link schedule example.

Figure. 2 shows a simple network configuration and available link scheduling for that network. Four timeslots are assembled into one slotframe, and four channeloffsets are available. Because of the variety of channels available, eight data transfers were completed within three timeslots. In the TSCH, dedicated cell is generally used for data stability. In a case of cell at [2, 3], a shared cell is used to allow multiple links to use one cell at a time, or one node to broadcast to multiple nodes.

How to configure the TSCH networks

Like the IEEE 802.15.4 standard, the TSCH distinguishes two types of nodes included in the network, Full Function Devices (FFDs) and Reduced Function Devices (RFDs).

In the case of FFDs, it is a node that has all functions defined in the standard and can perform as a coordinator of the network. RFDs can operate only some of the functions defined in the standard and cannot be as coordinators. The IEEE802.15.4 standard uses a control message called Enhanced Beacon (EB) to configure and maintain the network. When configuring the TSCH network at the first time, the FFD node, called as PAN coordinator, broadcasts the EB periodically at the shared cell. The EB message contains special information of the TSCH as follows:

- Synchronization Information: The information needed when a new node enters the network. It refers to the ASN value shared by all the networks.
- Channel Change Order: Passes a pre-specified random sequence of information to the new node so that it knows the channel change order.
- SlotOffset Information: Contains information about the length of the slotOffset and the number of slotOffsets that make up the slotframe.
- Information for initial communication: Contains information about the cells that are used for the first communication to join the network after acquiring information about the EB.

A node which wants to join the network turns on its radio and starts searching for available channels to receive the EB. As soon as the node receives the EB, MAC layer informs the upper layer about the receipt of the EB. The upper layer then allocates slotframe, timeslot, link according to the strategy and switches to the TSCH mode. After this process, the new node can communicate with other nodes already joined in the network. After all these steps, the newly joined node completes the process of participating in the network and broadcasts the EB around to inform the existence of the network to another nodes.

Routing protocol: RPL

IPv6 Routing Protocol for Low-power Lossy Networks (RPL) is designed for a network with low power and very noised sensors. The RPL is designed to support for many-to-one, one-to-many, and one-to-one traffic. Basic idea for the RPL is that nodes joining in the network create Destination Oriented DAGs (DODAGs) towards a central root node. These DODAGs are managed with a unique identification. The RPL can configure the DODAG for various purposes such as energy consumption, latency, network lifetime, etc. It is represented by Objective Function (OF). The OF is an equation including metrics to achieve goal. The OF may be selected by network administrator. Each node calculates RANK value using the OF. An advantage of the RPL is that each device can flexibly determine uplink path with the RANK value through simple OF operation.

The RPL has two modes about how to store routing information. One is non-storing mode in which each individual node cannot store routing information and only the root node can have the information. In this mode, peer-to-peer communication with other nodes must use the routing information in the root node. Another mode is a storing mode in which individual nodes can store routing information. This mode does not need to deliver to the root node because each node has routing information.

The RPL creates and maintains the upward routing path through the DODAG Information Object (DIO) message. The DIO message contains RPL INSTANCE, DODAGID, RANK, and DODAGVersion numbers. Each node collects DIO messages from neighbor nodes before joining the DODAG. Then, according to the policy of the OF, a highest priority neighbor is selected as parent. Then, it computes its RANK value based on the RANK value received from the parent.

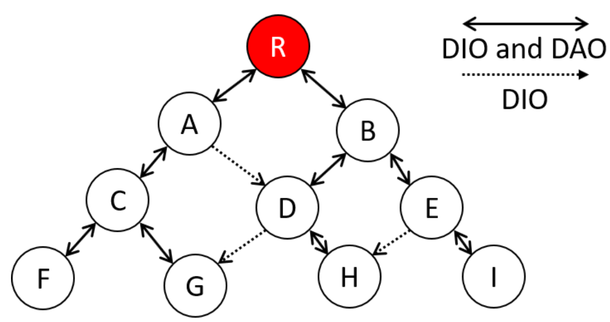

Figure 3. An example of the RPL

Figure. 3 shows the RPL routing protocol. When each node enters the network from the start node specified as root, it acquires neighbor information through the DIO messages. Among them, the node that best matches the OF strategy is selected as the parent node, and its RANK value is calculated based on the RANK value of the parent node. In the case of the node G in the figure. 3, the DIO message is received from the node C and the node D, but the parent node is selected as the node C. The DAO message is delivered through the node selected as the parent node, and finally the routing path for G is stored in the root. If you want to transfer data from H to G, communication is done by routing the data you want to transfer to the root, then receiving the routing information and moving the data to G.

Related Work

The EB broadcasting is an advertisement process to announce existence of the network. In the low-power wireless networks, it is common to prevent collisions between the EB transmissions to improve energy efficiency. However, in case of frequent re-joining due to dynamic network environment or mobility device, EB broadcasting strategy is needed to improve the EB reception success rate of the device. Therefore, in this paper, we focus on EB broadcasting strategy and look for related studies to reduce network participation time (or synchronization time).

De Guglielmo et al [3]. proposed a simple random-based EB broadcasting strategy. All nodes belonging to the network broadcast the EBs in the same slotOffset and channelOffset. In order to reduce the EB collisions, each node broadcasts the EB with probability. The authors of that paper first considered the EB collision as a factor affecting network participation rate, and therefore proposed a mechanism to reduce the EB collision by controlling the number of EBs broadcasted in a timeslot. However, the EBs are broadcasted in the same channel at only one cell without considering the multi-channel characteristics of the TSCH. It can be disadvantage because collisions may be occurred when a lot of nodes within the communication range broadcast the EB.

To complement the random-based strategy, Vogli et al [4]. proposed a concept of multi-slotframe extending slotframe and a Random Vertical (RV) filling and Random Horizontal (RH) filling mechanism using multiple channels. The multi-slotframe composed of multiple slotframe sets serves as one large repetition period. The RV mechanism allows nodes which can broadcast the EBs to randomly select one of the available channels to broadcast in the first slotframe of the multi-slotframe. It makes possible to broadcast the EBs on multiple channels at the same time, so increase the network participation rate of nodes. On the other hand, the RH mechanism fixes the EBs broadcasting channel to one and randomly selects a slotframe to broadcast the EBs. This increases the network participation rate by varying the EB broadcasting time. The same point between these mechanisms is to distribute the EBs broadcast to help new nodes join the network. However, it is not considered the situation where many nodes are involved. That is, the maximum number of channels defined in the TSCH standard is 16, and in a congested situation where there are 16 or more nodes at communication range, even if channels are divided communication interference due to the EB collisions may occur.

Vallati et al [5]. investigated the performance of the 6TiSCH network formation with the 6TiSCH Minimal Configuration and proposed a scheme which allocates/deallocates timeslots dynamically. They showed that transmissions of the EBs and RPL control messages can affect to the network formation performance.

Duy et al [6]. divided slotframe into advertisement plane and communication plane to avoid the EB collisions in congested situations. They proposed an algorithm that broadcasts the EBs simultaneously on multiple channels while controlling the number of EB messages using fuzzy logic according to the number of neighbor nodes in the advertisement plane consisting of five timeslots. It reduces the congestion of EB messages by adjusting the number of EBs according to changes in the network environment. However, due to the advertisement plane for broadcasting the EBs, the communication plane to exchange data is reduced. So, it is difficult to use in a situation where lots of timeslots are required because of frequent data exchange.

Karalis et al [7]. proposed the ATP which divides an advertisement slot into multiple subslots. It is possible because size of the EB is smaller than 40 bytes and link-layer broadcast is not acknowledged. But they used up to 5 nodes and did not evaluate in dense situation with lots of nodes.

Various proposals [8] and [9] emerged as ways to minimize the beacon collision. The author of [8] proposed a dynamic and configurable beacon timer called Bell-X. The time of beacon's timer is gradually increased or decreased, and the beacon transmission cycle is adjusted to prevent beacon collisions. The author of [9] initially proposed a dynamic beacons transmission system in which the transmission cycle of beacons changes according to the number of nodes joined during the network formation process. As a result of the simulation, the more nodes combined, the less join time.

Collision Minimization Beacon Scheduling

We describe a scheme to minimize collision beacon scheduling in this section. To apply this technology, we have also been able to broadcast EBs on Tx slot. To prevent EB collision of nodes, the time slot to broadcast EB is determined by using the result of the formula consisting of the combination of RANK and random value of the parent node. The information of the time slot determined at each node is transmitted to the ROOT through the DAO message to ensure the independence of the network. In this process, nodes that select the same timeslot are assigned different timeslots. Next, move the channel to broadcast EB.

Expansion of slot for sending EB to general Tx slot

In the IEEE 802.15.4e TSCH standard, the types of time slots are divided into the following four types: TX slot, RX slot, ADV slot, sleep slot. Typical TX and RX slots are used for data exchange and are allocated according to a scheduling policy. The ADV slot is used to broadcast the EB and is pre-allocated. The sleep slot is allocated when no data exchange occurs. In standard and previous studies, EB was broadcast only in the ADV slot. Therefore, the ADV slot for broadcasting the EB is preset in the slotframe. As a result, the EB was broadcast in the designated ADV slot, and it was difficult to apply the change to the situation of the network. When the number of nodes is equal to or less than the number of channels, the number of collisions of the EB does not increase because of the channel division technique, which is a characteristic of the TSCH. However, when the number of nodes is larger than the number of channels, the number of collisions increases in proportion to the number of nodes broadcasting EB within the communication coverage range. This is fatal for nodes that want to join the network. As the number of collisions increases, the number of EBs that can be received decreases. Therefore, the waiting time of the EB on the node increases. An increase in latency increases the energy consumption of the node, thus shortening the lifetime.

We changed the EB to broadcast in the TX slot. Therefore, EB broadcasts are not guaranteed to be broadcast more than once in a slotframe, unlike broadcasts in an ADV slot. To overcome this problem, we propose a method in which each node in the network allocates a specific time slot among the entire slotframe and broadcasts the EB there. Here, the node must select a slot according to a certain criterion. The timeslot used for general data exchange changes frequently depending on the type of data and the network conditions. In this case, it is advantageous to schedule the channel offset and timeslot to dynamically change because the receiving and transmitting nodes are certain. However, in the case of EB scheduling, it is impossible to know the existence of a node that wants to participate in the network. That is, the sender must transmit the EB without knowing the existence of the sender. So, it is almost impossible to dynamically distribute the scheduling of EBs. Therefore, the proposed scheduling method allocates one Timeslot for each node by using RANK value and random value. This mechanism can prevent network congestion due to EB collision because it is assigned a timeslot different from neighboring nodes in the network. In addition, since the channel movement is performed in accordance with the increase of the ASN, a node joining the network can join the network at a maximum number of 16 or less, regardless of which channel the node is listening to. It is possible to calculate the maximum network join time differently from other randomly based dissertation techniques.

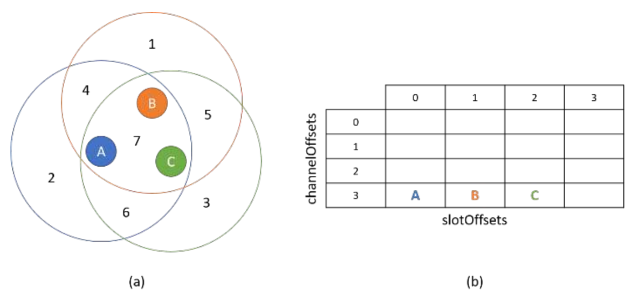

Figure 4. An example of increasing the number of EB broadcast frequencies according to node density

Another advantage of this mechanism is that as the number of nodes capable of communicating in the network range increases, the network join time of the node that wants new joining of that part decreases. Figure 4 (a) shows three nodes participating in the network with overlapping communication ranges. Figure 4 (b) shows the timeslots assigned by nodes A, B, and C based on RANK. Since each node broadcasts EB at different timeslots, there is no possibility of collision between EBs. In Figure 6 (a), the broadcasted EB of A node is broadcast within the range of 4, 2, 6, and 7, respectively. In the same way, the B node emits EBs in zones 1, 4, 5, and 7, and the C node broadcasts EBs in zones 5, 3, 6, Here, Zone 2, Zone 3, and Zone 1 are ranges in which EBs of each node are broadcast one by one. Therefore, the maximum network participation time in this area is the time to join after traveling through all 16 channels. Next, for Zone 4, Zone 6, and Zone 5, the EBs of the two nodes overlap. Therefore, a total of 2 EBs are transmitted per channel, and the maximum network join time is less than the maximum network lead time mentioned above. Finally, in the case of zone 7, all the EBs of nodes A, B and C are broadcasted. Therefore, the maximum network time is further shortened.

Rank based Timeslot Duplicate Prevention Mechanism

This chapter describes how each node selects its own timeslot using the RANK used in the RPL protocol. To recycle the limited slotframe, we first discuss the strategy of partitioning. This method uses the N_{depth_total} value indicating how many nodes on the RPL can communicate with based on the root node. Equation 2 is to find the number of timeslots per section. Equation 3 represents an expression for obtaining the starting point of a section by using the number of timeslots per section obtained by the Equation 2. This starting point serves as the basis for the section for timeslot recycling.

$$N_{section\_timeslot} = \frac{N_{slotframe}}{N_{depth\_total}}, \quad (2)$$

$$N_{section\_start\_timeslot}(i) = \frac{N_{section\_number}(i)}{N_{section\_timeslot}} \quad (3)$$

The node joining the network uses N_{depth_total} and $N_{slotframe}$ contained in EB to obtain $N_{section\_timeslot}$. Based on this, we select Timeslot by itself using Algorithm 1.

Algorithm 1. Timeslot selection algorithm
1: receive R(P) and N_{depth_total} from preferring parent table
2: R(N) = R(P) + INCREASE
3: OFFSET = R(N) % $N_{section\_timeslot}$
4: Timeslot = K[N_{depth}] + OFFSET

N_{depth} indicates whether the current node is connected via a certain node from the root node. This value is incremented by 1 based on the parent node. The RANK value of the Node is obtained as a result of the OF calculation. OFFSET specifies the position in the section by dividing the R (N) value obtained in the second line by $N_{section\_timeslot}$. Next, select section on line 4. If you put N_{depth} in the K function, it prints the value of the start timeslot that matches the section. Add it to OFFSET to determine the final Timeslot. Because the section varies depending on the N_{depth} of the node, the section can be recycled every N_{depth}. This is useful when applied to large networks.

Timeslot duplication prevention mechanism using RPL's DAO and DAO ACK

The previously proposed RANK-based time slot selection technique is a mechanism in which the node itself determines its own time slot using the parent's RANK value. However, the preceding mechanism is problematic in two cases: First, the number of nodes to be allocated is larger than the size of the timeslot per section. The second problem arises when the values of OFFSET are the same and are assigned to the same timeslot, and the timeslot selected by the node itself is duplicated. These two problems occur when nodes join the network frequently. Therefore, this chapter discusses the mechanisms that complement these two problems.

The RANK-based timeslot mechanism discussed above uses the RPL technology to form a network. We transmit the timeslot allocation information of the nodes in the network to the Root node using DAO message, which is the control message used for network formation.

Algorithm 2-1 shows the Node-pointed Timeslot duplication prevention mechanism. After setting the Timeslot using the method shown in the previous section, send Timeslot and $M_{section}(N_{section} = N_{depth} \% N_{depth\_total})$ to DAO message to complete the network configuration. Then, the DAO ACK is received by the Root Node. If it is the same as the previously selected Timeslot, EB is broadcast at that location. If it is different, EB is broadcasted in the changed Timeslot. Line 1 sets Timeslot with the RANK-based Timeslot selection algorithm discussed earlier. Lines 2 and 3 contain the Timeslot and the depth number to which it belongs in the DAO, send it to the Root node, and wait for the ACK. Line 4 ~ 5 receives the ACK and changes to the new Timeslot sent by the Root node if the timeslot does not overlap with the one calculated by itself. If the duplicated Timeslot comes, it confirms the Timeslot that was set first and broadcast EB.

Algorithm 2-1. Timeslot duplicate correction algorithm (Node perspective)

```
1: Timeslot = K[N_depth] + OFFSET
2: make the DAO message including Timeslot and N_section
3: wait the DAO Ack
4: receive the DAO Ack
5: change Timeslot or determine Timeslot
```

Algorithm 2-2 shows the Timeslot redundancy correction algorithm from the root node viewpoint. Root node receives DAO and sends DAO ACK.

The root node receives a DAO message from each node in the network. Check the depth of the node assigned to the Timeslot from the existing network information. If there is no node assigned to the transmitted Timeslot, the value of timeslot is sent as it is. If there is an assigned node, select an empty Timeslot and send it to the message. The first line receives a DAO message from the Node. Line 2 is the process of verifying that there is a duplicate Timeslot node in the network using the Timeslot and section information in the DAO. Lines 3-8 indicate the process of determining if a Timeslot is used, if it is used. Line 5 compares the depth of the node using the existing Timeslot using Timeslot to the depth of the node that is currently using it. If it is the same depth, assign another Timeslot. If it is not the same depth, it is regarded as Timeslot reuse, and put the default message in the DAO as shown on line 8. Line 10 indicates that an existing message is sent in a DAO when Timeslot is not used. Finally, the DAO ACK is sent to the node that sent the DAO. Here, the method for determining the new timeslot in line 6 can be made through various strategies.

Algorithm 2-2. Timeslot duplicate correction algorithm (Root perspective)

```
1:  receive the DAO message
2:  check TimeslotList[Timeslot] and section
3:  if (Timeslot is used)
4:    check N_depth
5:    if (depth_before == depth_now)
6:        include < new timeslot >
7:    end if
8:    include < default message >
9:  else
10:   include < default message >
11: end if
12: send the DAO Ack
```

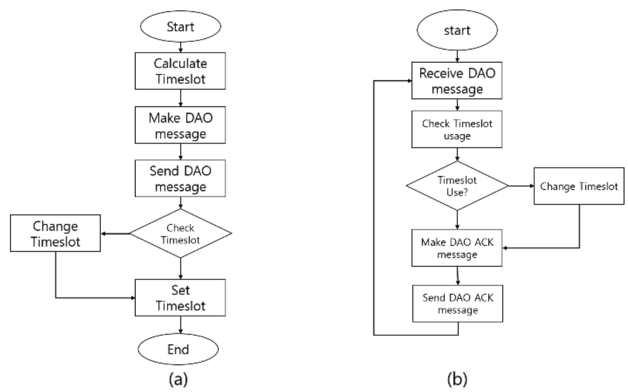

Figure 5. Timeslot duplicate correction algorithm flow diagram. (a): node, (b): root

Figure 5 shows the flow diagram of the previous two algorithms. (a) is the node view for network participation, and (b) is the root node view. (a), the flow diagram starts when the node belonging to the network receives the broadcast EB. Compute Timeslot with the algorithm described above, create a DAO message, wait for ACK after sending it to the Root node. If the Timeslot in the ACK from the Root node matches the timeslot calculated by the ACK, it starts EB broadcasting as it is, changes its Timeslot, and starts broadcasting. (b), it is always ready to receive the DAO. When a DAO is received, it compares the specified Timeslot of the entire network node of the Root with the Timeslot received from any node. If you do not use any node on the network, it will send the value back to ACK and, if you are using it, send the specified Timeslot information to ACK using a different strategy.

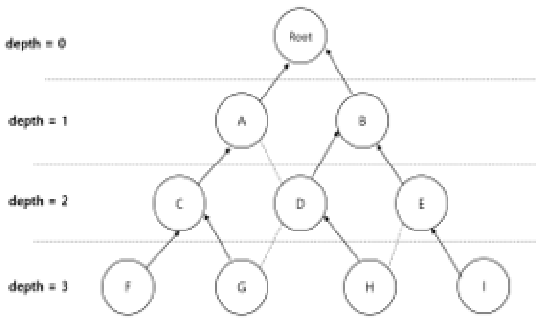

Figure 6. RANK-based Timeslot selection mechanism example network diagram

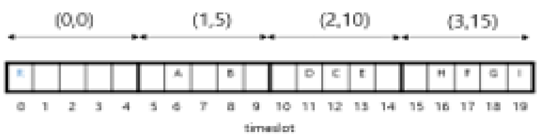

Figure 7. RANK-based Timeslot selection mechanism timeslot example

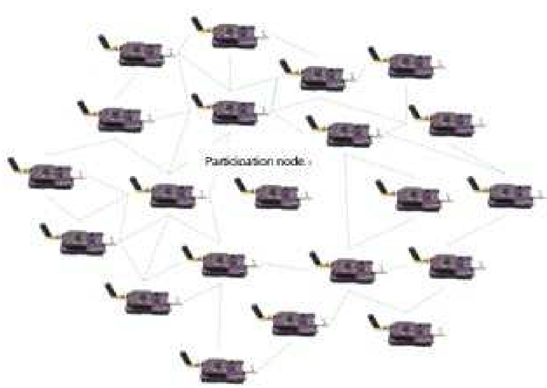

Figure 8. Experimental network configuration

Figure 6 and Figure 7 show examples of RANK-based time slot selection mechanism and redundancy prevention mechanism. The depth is divided into 4 depths, and the beginning of each depth consists of 0, 5, 10, and 15 values. The length of the slot frame is 20. For the first depth, timeslot starts at 5. Nodes A and B receive the EB message from the root and join the network. One higher depth is determined than the upper depth value. Using the formula, we discussed earlier, we calculate the Timeslot. This is recorded in the second 4th Timeslot of Section 1 to identify the time slot. The determined value is stored in the DAO message, sent to the root, and broadcasted at that location after the Timeslot confirm operation is completed. Next, since C, D, and E belong to depth 2, the timeslot starts at 10 and is calculated based on the parent node's RANK value. The following sections apply in the same way. When a new node of F, G, H, or I is entered, the section changes back to zero. In this case, you can use channel 0, which is used by root. That is, a node having an enough distance and outside the communication distance can reuse the time slot.

Methods and Experimental

Parameter	
Run number	100
Mote	Open-Mote
the number of nodes	2-30
the number of channels	16
slot length	15ms
slotframe length	101
multi slotframe	10 slotframes

Table 1. Parameter

All experiments conducted in this study were prepared on an OpenWSN [11] stack, and all the experiments were implemented using Open-Mote [10]. Figure 8 shows the experimental network configuration. Table 1 shows parameter of implementation setup. A total of 30 Open-Mote [10] experiments were conducted. All 30 nodes are within communication range and maintain the DODAG according to the RPL protocol. The participating node selects one of the 16 channels for network entry at the network center and awaits an EB reception. The timeslot length is 15ms, and the slot frame consists of 101 timeslots. A multi-slot frame consists of ten slot frames.

The network participation time is measured by calculating the minimum network joining time after the power of the node for network participation is turned on and the ASN value is converted into seconds. The average network join time represents an average of 100 values, and the maximum network join time represents the maximum value among the 100 times of measurements.

We compared the proposed LCBS algorithm with the random-based beacon scheduling algorithms such as RV, RH, and Rapid presented in the relation section with the average network subscription time, maximum network participation time, the proportion of EB slots in the total number of timeslots.

Results

Comparison of average network join time

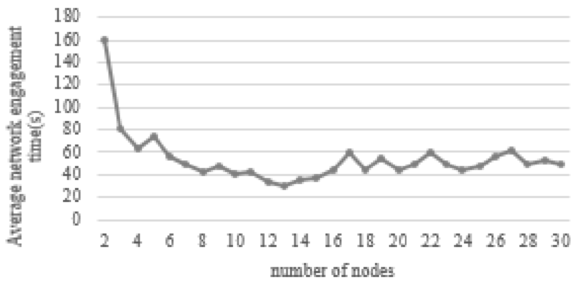

Figure 9. Random-based beacon scheduling average network joining time

Figure 9 shows the average network joining time of RV, which is a kind of random-based beacon scheduling. The vertical axis represents the average network participating time, and the horizontal axis represents the nodes participating in the network. As the number of nodes participating in the network ranges from 1 to 13, as the number of nodes increases, the number of EBs broadcast on the channel increases, so the average network participation time decreases naturally. However, when the number of network participating nodes is more than 14, the participation time of the network is gradually increasing. This is because the nodes participating in the network randomly select channels to broadcast EBs, and when the number of nodes increases, the number of conflicts between EBs increases because the number of nodes selected for the overlapping channels increases.

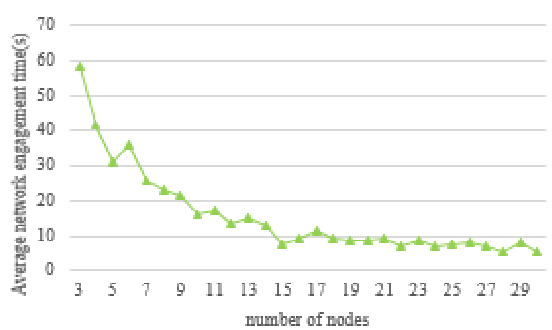

Figure 10. Collision minimization beacon scheuling average network joining time

Figure 10 shows the average network participating time of our proposed collision minimization beacon scheduling. Unlike the results shown in Figure 11, the average network participation time decreases steadily as the number of nodes increases. We applied the strategy of broadcasting the EB by separating Timeslot. As a result, there is almost no collision between EBs, and the number

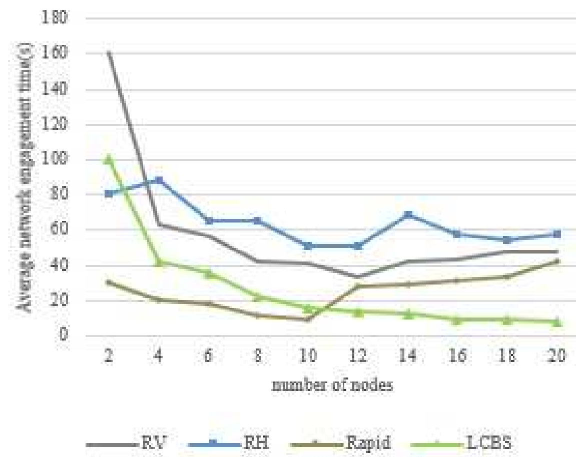

Figure 11. Average network joining time

of EB transmissions per slotframe increases proportionally to the number of nodes in the network. The average network participating time of the nodes that want to join the network has steadily decreased. Especially, from the point where the number of nodes where channel saturation occurs is more than 17, it is different from Fig. 10. We can expect this trend to continue until the slot frame is saturated. In this case, if we use the section reuse scheme through slotframe partitioning, we can use collision minimization beacon scheduling method regardless of the slotframe length.

Figure 11 shows a graph of the average network participation time of the conventional random-based beacon scheduling scheme and the scheduling scheme presented in this paper. In the case of RV, the average network participating time is steadily decreased to 12 nodes, and it decreases to 34 seconds maximum. However, the network participation time tends to increase slightly from 12 or more. The reason for this is that EBs are randomly selected from among 16 channels and EBs collide. In the case of RH, the participation time was continuously decreased to 12, and the time was reduced to 50 seconds. The reason why RH takes network participation time longer than RV is because RH has more than 10 randomly selectable slots, which means more conflicts and longer average participation time. In the case of the Rapid technique, the average network participation time is the shortest in 2 ~ 10. The reason for this is that, between 1 and 2 EBs, three EBs are broadcast between 5 and 3 and 6, and one EB is broadcast continuously. Therefore, it can be said that the joining time is faster because one node emits several EBs. However, since the number of EBs broadcasting one EB per node is decreasing, the number of network participation time is gradually increasing due to collision between EBs and RVs after 12th. In the case of collision minimization beacon scheduling (LCBS) proposed in this paper, we can see that the average network participation time decreases continuously as the number of nodes increases. Experimental results show that the number of nodes is more than 25 ~ 27 seconds faster than the average network participation time of other techniques. The gap becomes more different as the number of nodes increases. This difference occurs because the nodes participating in the network broadcast the EB by dividing the Timeslot, so there is no conflict with EB. It can be expected that the scheduling technique presented in the paper shows a better effect in a factory environment where electronic devices and equipment are concentrated.

Comparison of maximum network participation time

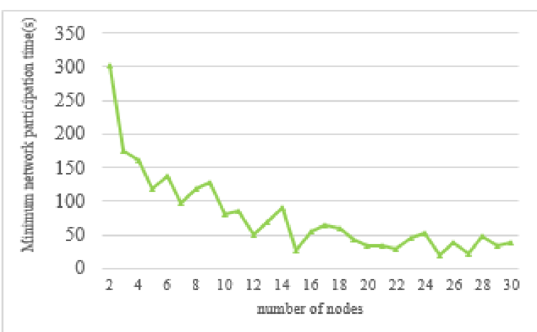

Figure 12. Maximum network joining time with the Collision Minimization Beacon Scheduling

Figure 12 is a graph showing the result of measuring the maximum value of the collision minimization beacon scheduling network participation time that we proposed. Since each timeslot is assigned to a node, the maximum network participation time decreases as the number of nodes in the network increases. A special point is that when the number of channel nodes is 16 or more, the value is not changed any more but is changed between 30 and 50 seconds. This is because the value no longer shrinks due to the saturation of one section. Figure 13 compares the maximum value of the network participation time

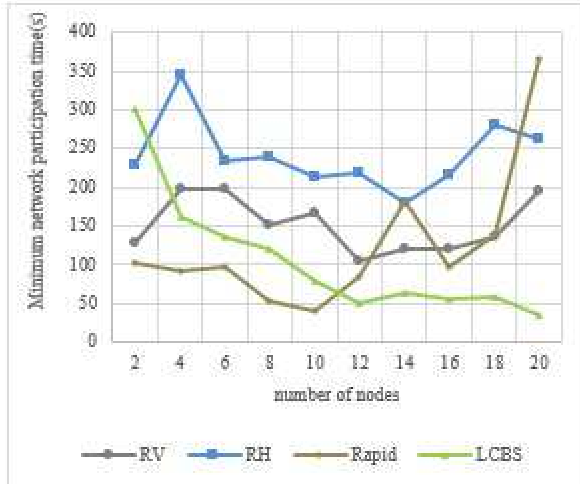

Figure 13. Comparision of network joining time

between the conventional random channel selection-based technique and the proposed paper technique. Conventional random channel selection-based techniques change the maximum value without special tendency because the channel is selected at random. This can serve as a variable in terms of network design. Since it is possible to measure the joining time only through experiments, it is necessary to obtain the value through multiple measurements each time. However, because the random selection of the nodes belonging to the network instantaneously, we cannot trust the value through the measurement. However, in the case of the collision minimization beacon scheduling proposed by us, the maximum network participation time tends to decrease when the number of nodes increases, and the average joining join time between 30 and 50 seconds can be predicted. This can contribute significantly to the stability of the network design.

Comparison of the proportion of EB slots in the total number of timeslots

Some of the Timeslots that make up the slotframe are used to send EBs, and the rest are used to exchange data. As the number of slots used to transmit EBs in the entire Timeslot is smaller, the number of slots to which data can be transmitted is increased, and the data rate of the entire network is increased. In this experiment, Rapid [6], a comparison algorithm, compares our algorithm with the technique of changing the number of EBs according to neighbor nodes. The experiment was divided into 5 nodes and 20 nodes.

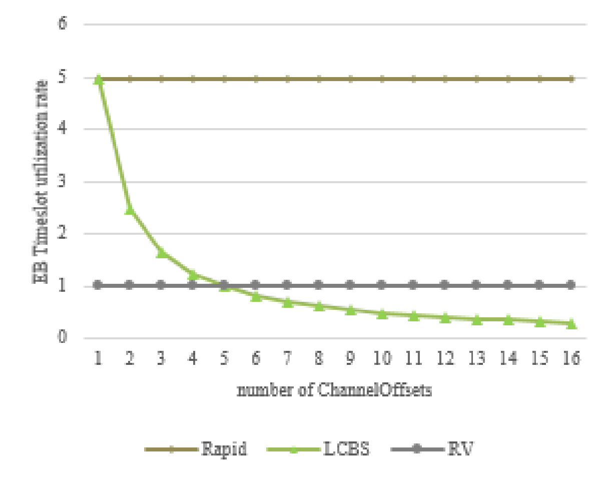

Figure 14. EB slot utilization rate with 5 network nodes

First, the experiment results of the EB slot utilization when the neighbor nodes are 5 are shown in Figure 14. The horizontal axis of the graph represents the number of channeloffset, and the vertical axis represents the EB Timeslot utilization rate. This shows the ratio of the EB slot to the total number of slotframes. In the case of Rapid, always use 5 ADV slots to transmit EB. channeloffset * 5 timeslot slot, always keeping a 4.95% share. Since RV uses one ADV slot as in the standard, it is used for EB by the number of 1 timeslot * channeloffset regardless of the number of channeloffset. Therefore, EB Timeslot share of 0.99% is always maintained. However, in the case of LCBS, EB broadcasts in the same slot as the number of nodes participating in the network. Therefore, if the channeloffset increases and the number of available timeslots increases, the share of the EB slot decreases. This shows that the EB slot usage rate is lower than Rapid in all cases except for one channeloffset, and EB slot usage rate is lower than RV for channeloffset of 6 or more.

Figure 15 shows the usage rate of 20 EB slots of the entire network node. RV slot utilization and Rapid slot utilization are still maintained at 0.99% and 4.95%, as in the previous experiment. However, in case of LCBS, slot is allocated according to the number of nodes, so it is 20% for one channeloffset, and lower than that for Rapid in case of four. In case of 16, it shows the same EB slot usage rate as RV. If you use less channeloffset, there is not enough slot for data exchange in LCBS. This is because one EB slot is assigned to each node. However, since the number of timeslots increases with the use of channeloffset more, the utilization rate of the EB slot of LCBS becomes lower as the number of channeloffset increases. In the case of the strategies of the previous

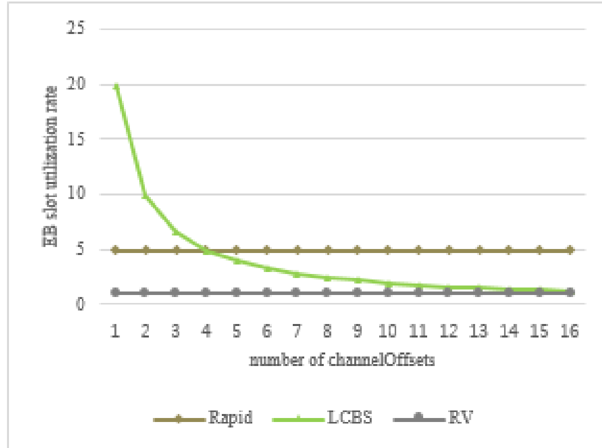

Figure 15. EB slot utilization rate with 20 network nodes

paper, since the EB slot is fixed to the advance slot in advance, unnecessary EB slot is generated. This may lead to a waste of a slot in a large-scale sensor network in which data exchange is frequent, so data exchange may not be performed properly.

In this experiment, we can see the network in which LCBS is used efficiently. It is optimized in a large-scale sensor network environment where more nodes than 16 channels which are standard channels of TSCH exist in the network.

Discussion

In this paper, we propose a beacon scheduling plan that minimizes conflicts in TSCH-based high density IoT environment. The IEEE 802.15.4e TSCH standard does not specify how the network is constructed. To solve this problem, other papers randomly selected channels to reduce network participation time. However, because these methods are based on randomness, the expected convergence time is unpredictable, and network congestion occurs due to conflicts between EBs.

We propose that LCBS algorithm presented a method of assigning to timestlot using RANK values of RPL. The proposed performance is implemented in OpenWSN Stack, and all the experiments is implemented in Open-Mote. A total of 30 Open-Mote tests is conducted, and each of the 30 nodes is in a communicable range, maintaining DODAG in accordance with the RPL protocol.

According to the results obtained, the collision-minimizing beacon scheduling technique presents in this paper measured that the average network joining time is about 25 seconds to 27 seconds faster in a network-intensive environment, and the network participation time in this paper also decreases by 30 seconds to 50 seconds when the number of nodes increase compared to the existing methods.

This experiment have limitations where resources through EB can be wasted by focusing only on the role of EB to minimize conflicts with only channel connection information on nodes. For future work, the EB will be expanded and scheduled throughout the slotframe, adding a method of linking with scheduling for actual DATA exchanges.

Declaration

Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Korea Institute of Energy Technology Evaluation and Planning(KETEP) and the Ministry of Trade, Industry & Energy(MOTIE) of the Republic of Korea (No. 20182020109660).

Authors Contribution

All authors contributed to the design and analysis of the research, to the simulation results, and to the writing of the manuscript. All authors read and approved the final manuscript.

Abbreviations

WSN: Wireless Sensor Network
TSCH: Time Slot Channel Hopping
IoT: Internet of Things
MAC: Media Access Control
RPL: Routing Protocol for low-power Lossy Network
EB: Enhanced Beacon
DODAG: Destination Oriented Directed Acyclic Graph
DIO: DODAG Information Object
DAO: Destination Advertisement Object
ASN: Absolute Slot Number
FFDs: Full Function Devices
RFDs: Reduced Function Devices
PAN: Personal Area Network
OF: Objective Function

References

1. IEEE 802 WORKING GROUP, et al. IEEE Standard for Local and Metropolitan Area Networks—Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs). IEEE Std, 2011, 802: 4-2011.
2. Thubert P, Brandt A, Hui J, RPL: IPv6 routing protocol for low-power and lossy networks, IETF RFC 6550, March2012. Google Scholarrapid.
3. DE GUGLIELMO, D, et al. A performance analysis of the network formation process in IEEE 802.15. 4e TSCH wireless sensor/actuator networks. In: Computers and Communication (ISCC), 2014 IEEE Symposium on. IEEE, 2014. p. 1-6.
4. VOGLI, E, et al. Fast join and synchronization schema in the IEEE 802.15. 4e MAC. In: Wireless Communications and Networking Conference Workshops (WCNCW), 2015 IEEE. IEEE, 2015. p. 85-90.
5. C. Vallati, S. Brienza, G. Anastasi and S. K. Das, "Improving Network Formation in 6TiSCH Networks," in IEEE Transactions on Mobile Computing, vol. 18, no. 1, pp. 98-110, 1 Jan. 2019.
6. DUY, T; DINH, T; KIM, Y. A rapid joining scheme based on fuzzy logic for highly dynamic IEEE 802.15. 4e time-slotted channel hopping networks. International Journal of Distributed Sensor Networks, 2016, 12.8: 1550147716659424.
7. A. Karalis, "ATP: A Fast Joining Technique for IEEE802.15. 4-TSCH Networks," 2018 IEEE 19th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), Chania, 2018, pp. 588-599.
8. Vera-Pérez, J.; Todolí-Ferrandis, D.; Silvestre-Blanes, J.; Sempere-Payá, V. “Bell-X, An Opportunistic Time Synchronization Mechanism for Scheduled Wireless Sensor Networks.” Sensors 2019, 19, 4128.
9. A. Kalita and M. Khatua, "Faster Joining in 6TiSCH Network using Dynamic Beacon

Interval," 2019 11th International Conference on Communication Systems & Networks (COMSNETS), Bengaluru, India, 2019, pp. 454-457.

10. Open-Mote [Online]. Available: http://openmote.com
11. Berkeley's openwsn. [Online]. Available: http://openwsn.berkeley.edu.

Figures

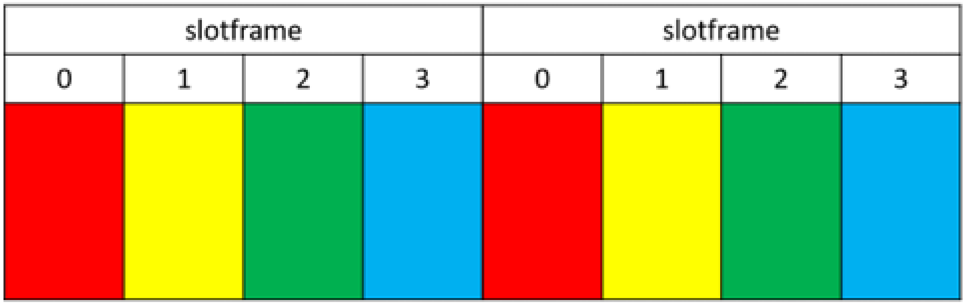

Figure 1

The TSCH slotframe structure.

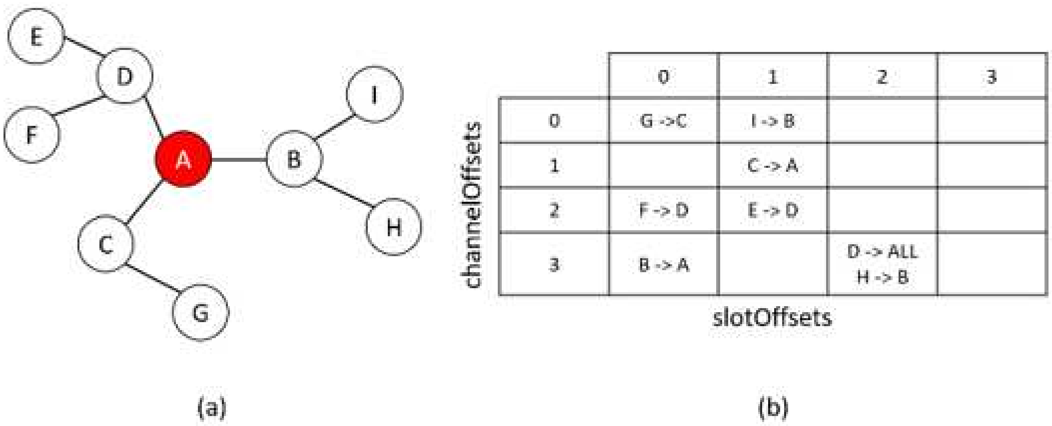

Figure 2

(a) TSCH network topology and (b) link schedule example.

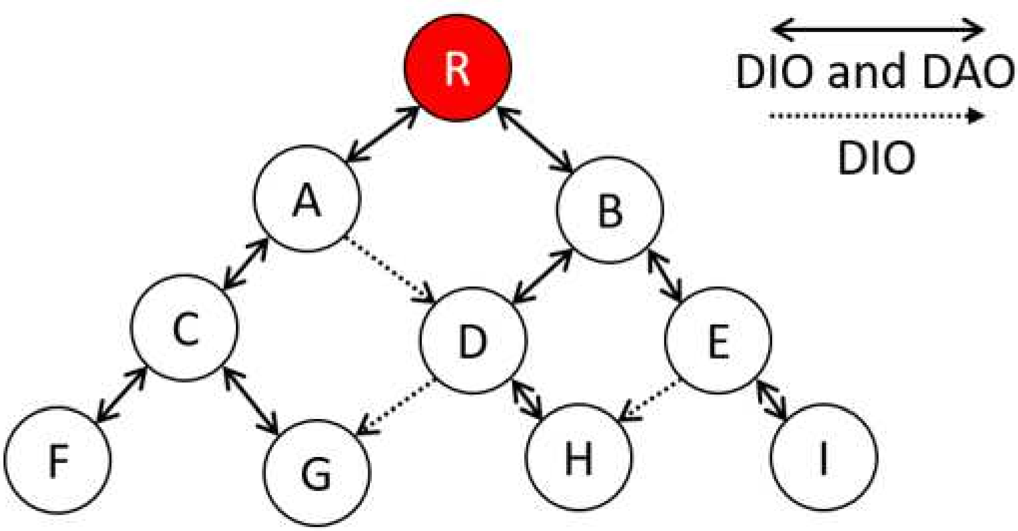

Figure 3

An example of the RPL

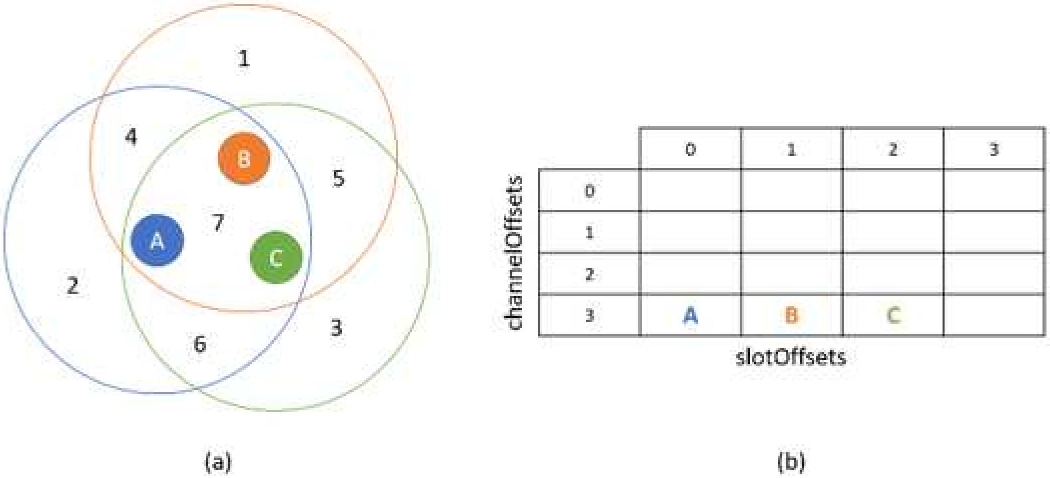

Figure 4

An example of increasing the number of EB broadcast frequencies according to node density

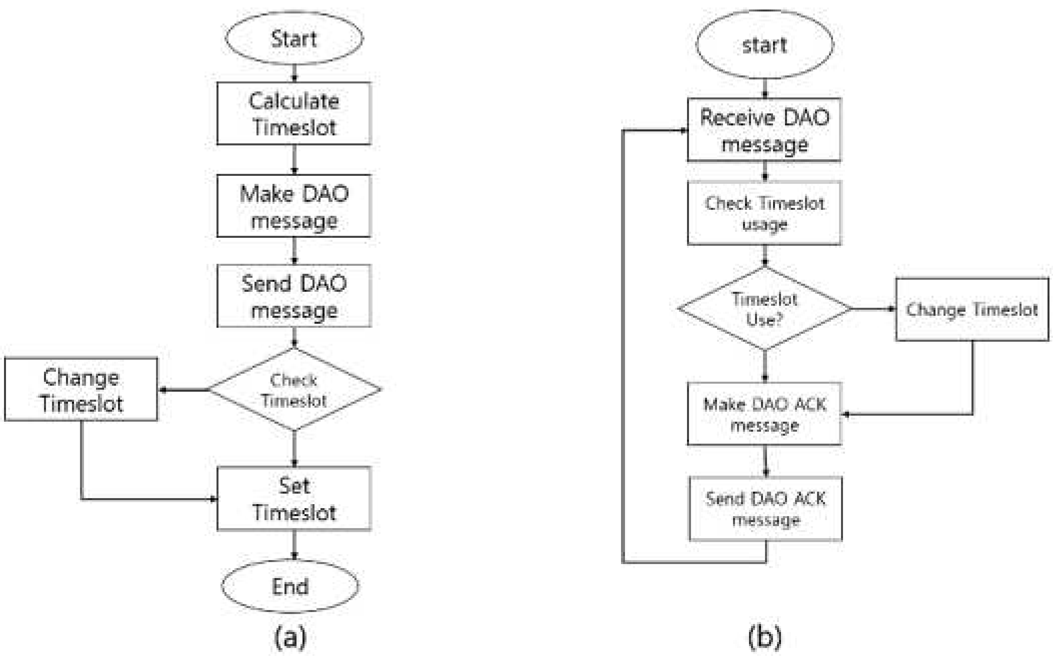

Figure 5

Timeslot duplicate correction algorithm flow diagram. (a): node, (b): root

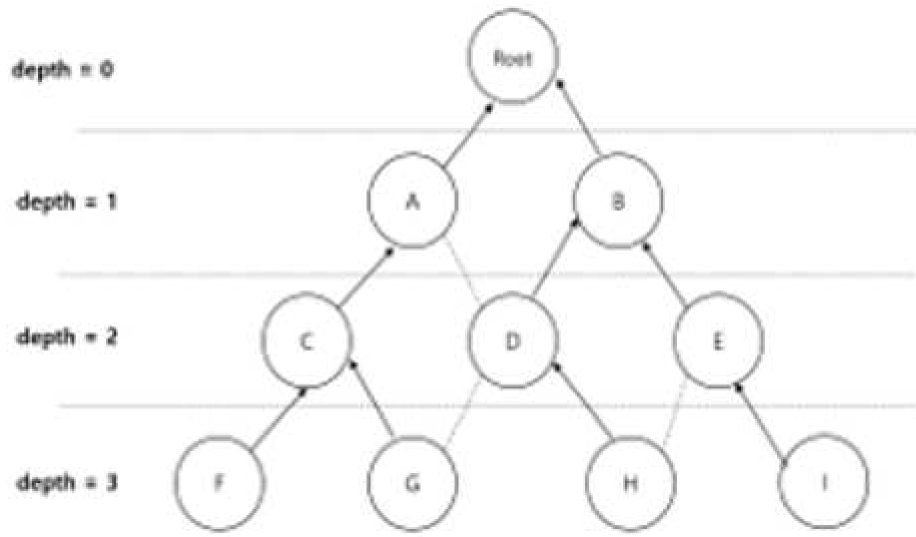

Figure 6

RANK-based Timeslot selection mechanism example network diagram

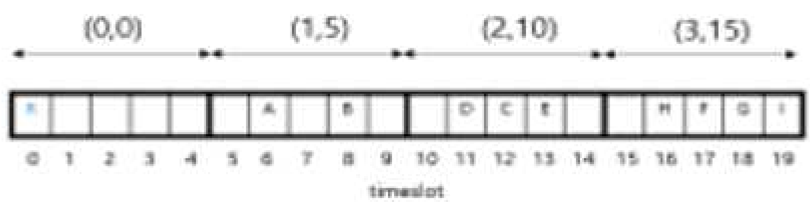

Figure 7

RANK-based Timeslot selection mechanism timeslot example

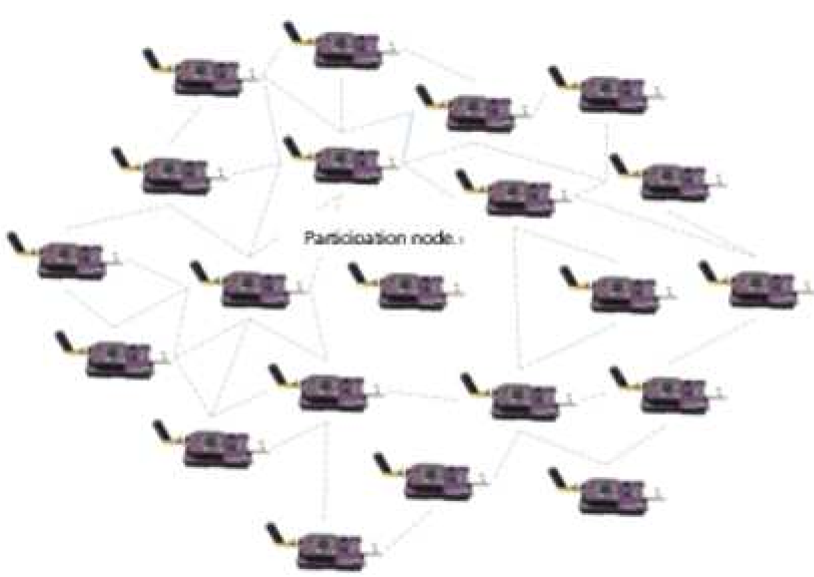

Figure 8

Experimental network configuration

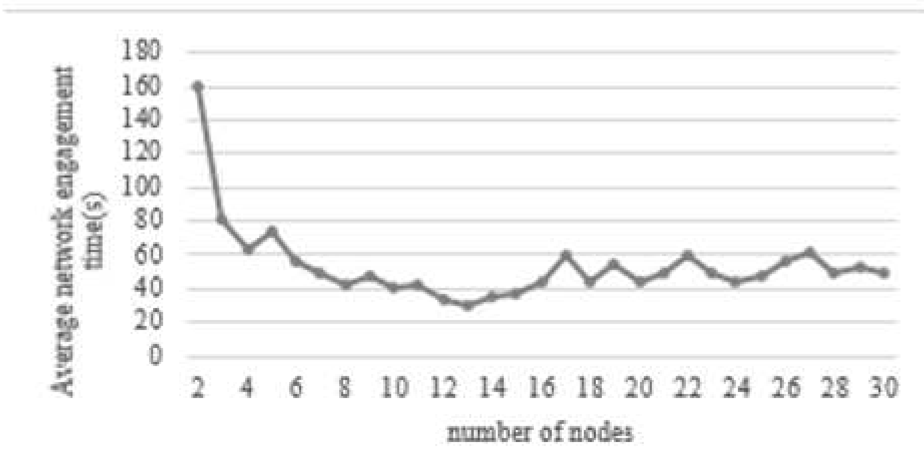

Figure 9

Random-based beacon scheduling average network joining time

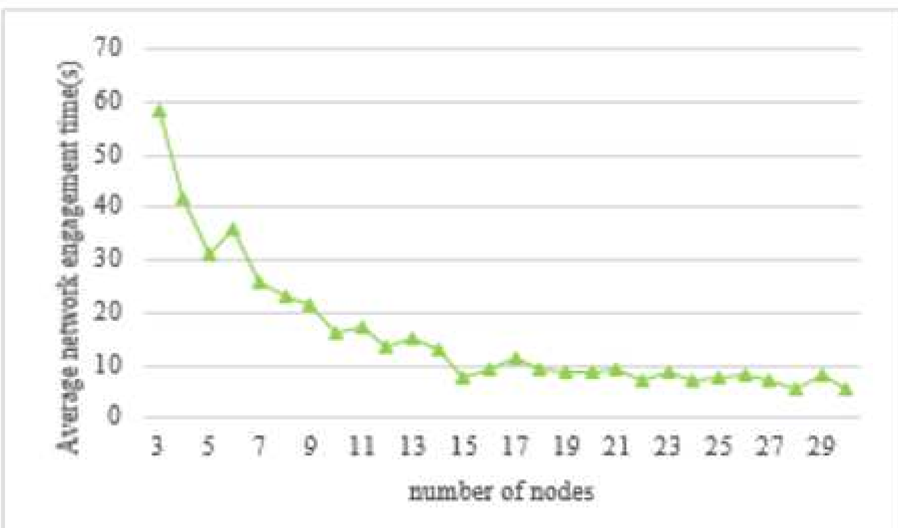

Figure 10

Collision minimization beacon scheuling average network joining time

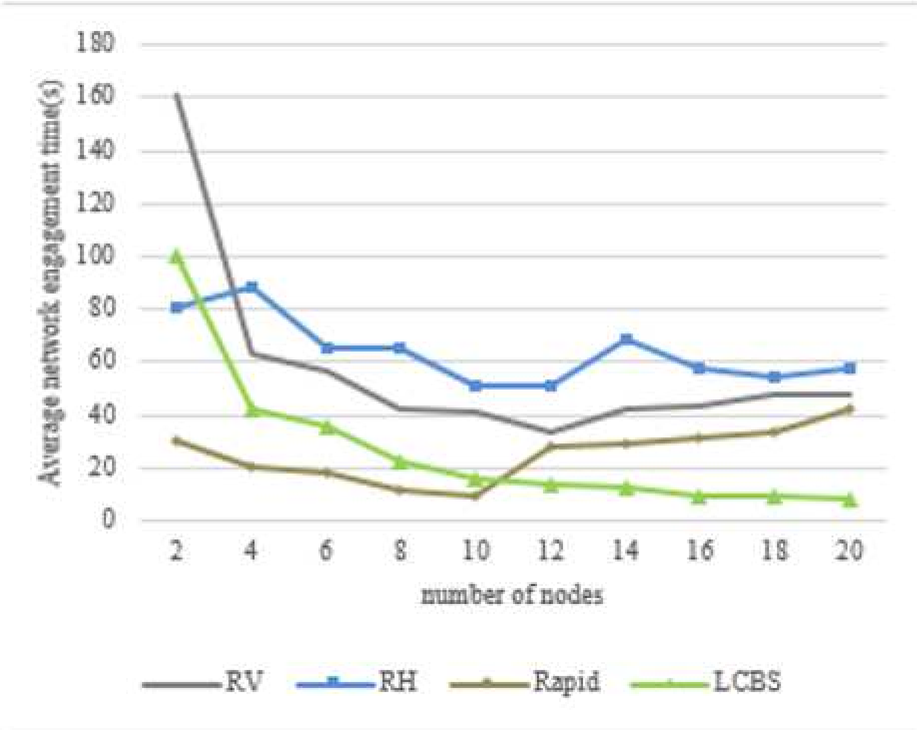

Figure 11

Average network joining time

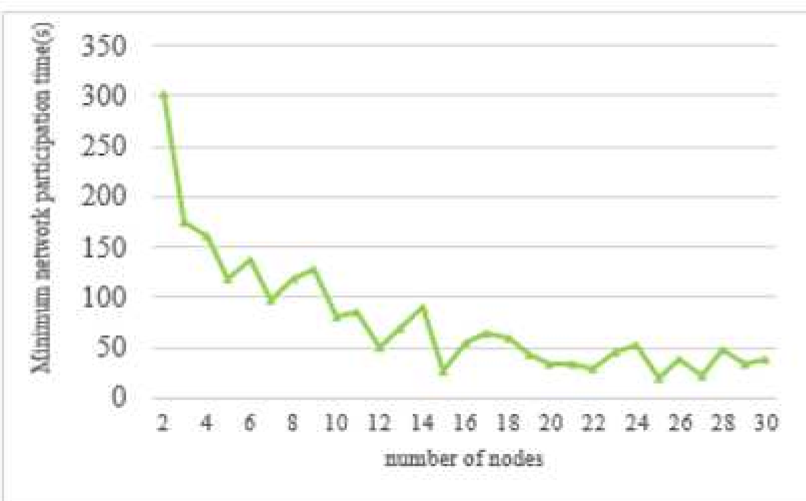

Figure 12

Maximum network joining time with the Collision Minimization Beacon Scheduling

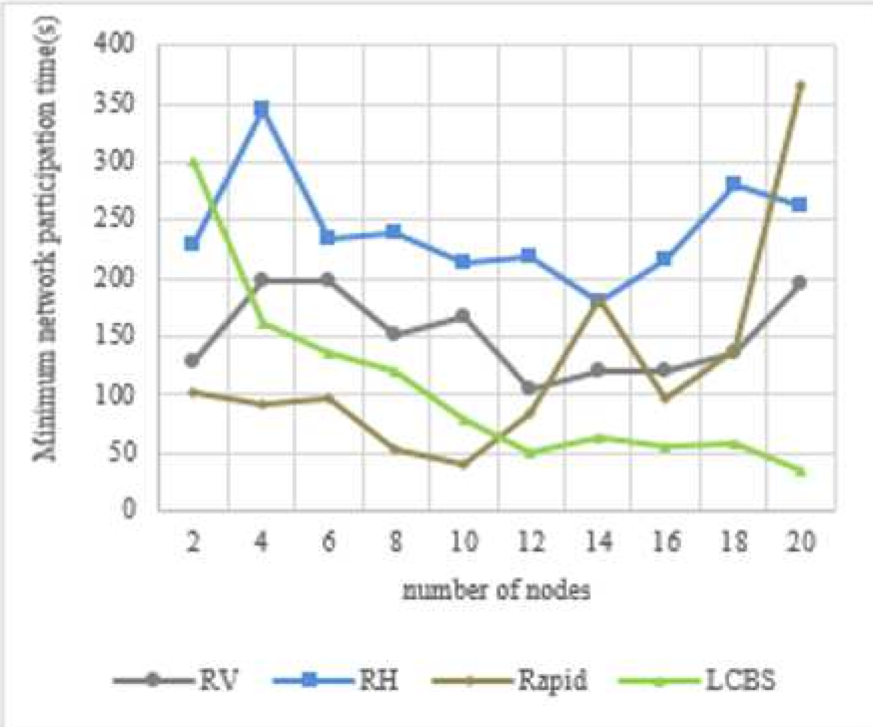

Figure 13

Comparision of network joining time

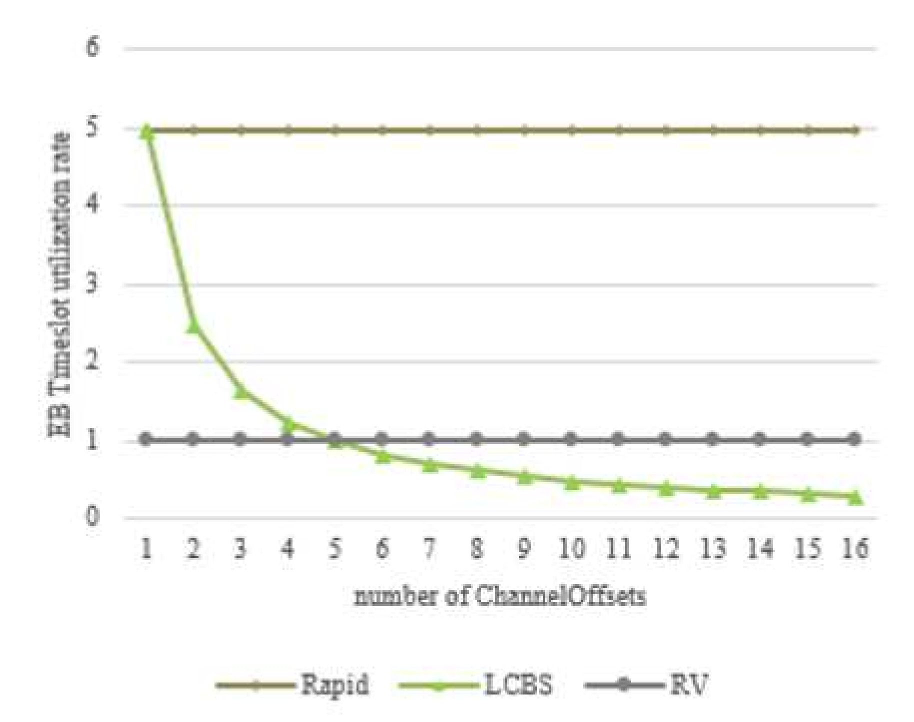

Figure 14

EB slot utilization rate with 5 network nodes

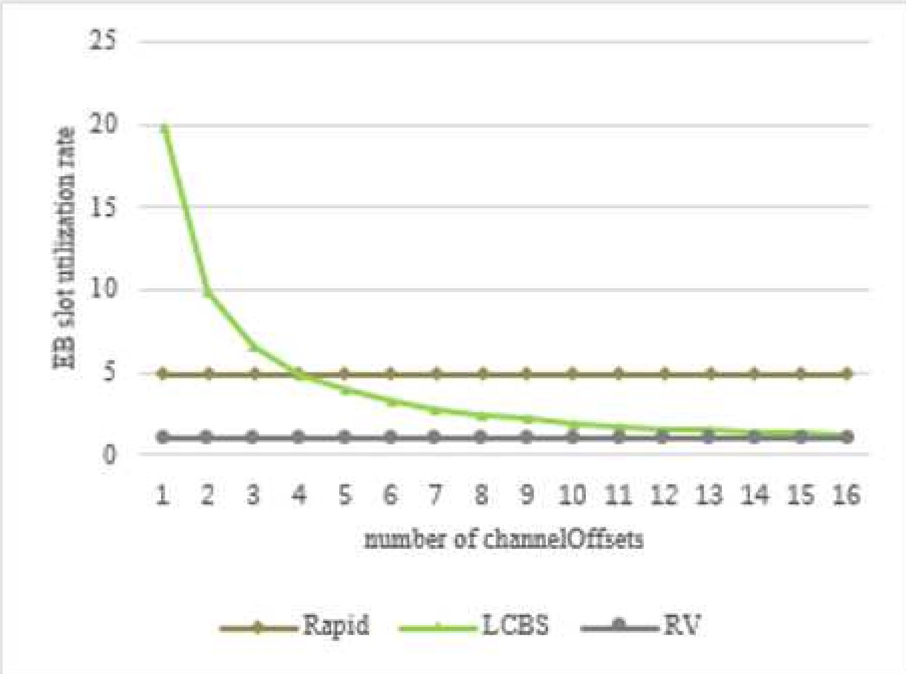

Figure 15

EB slot utilization rate with 20 network nodes