Faster Joining in 6TiSCH Network using Dynamic Beacon Interval

Alakesh Kalita
Computer Science and Engineering
Indian Institute of Technology Guwahati
E-mail: alakesh.kalita1025@gmail.com

Manas Khatua
Computer Science and Engineering
Indian Institute of Technology Guwahati
E-mail: manaskhatua@iitg.ac.in

***Abstract*—Internet of Things (IoT) in industrial environment (aka IIoT) is playing a crucial role in reshaping the current industry. IPv6 network on the top of Time Slotted Channel Hopping (TSCH) MAC protocol, commonly known as 6TiSCH, is one of the emerging protocol for IIoT. Industrial use of IoT demands high reliability, deterministic latency and high scalability with energy efficiency. In this paper, initial network formation procedure in a 6TiSCH network is assessed analytically. Analysis reveals that the performance of a 6TiSCH network degrades when a new node joins the network. To overcome this performance degradation, we propose a dynamic beacon interval scheme in which the interval varies with the number of joined nodes during network formation. We compare the performance of the proposed scheme with the minimal configuration standard in which static beacon interval is used. Simulation results show that the proposed scheme offers better performance in terms of joining time during network bootstrap.**

***Index Terms*—Industrial IoT, 6TiSCH, Network formation**

I. Introduction

In present times, Internet of Things (IoT) is broadly used in various application domains, and one of such domains is the industrial sector. The Industrial IoT (aka IIoT) is intended to reshape the current industry with the integration of sensors and actuators for real-time process monitoring and automation. IIoT demands high reliability, deterministic latency, less energy consumption and scalable network. As the IEEE 802.15.4 standard, specifically desingned for wireless sensor networks (WSNs), fails to offer such requirements, in 2008, 802.15.4e was introduced by IEEE. In IEEE 802.15.4e, five new MAC protocols (called MAC behaviour modes) were introduced to support specific requirements of application domains and few generic requirements. One of such MAC protocols is the Time Slotted Channel Hopping (TSCH), which provides time division multiple access with channel hopping allowing more number of parallel communications, and can achieve high reliability with low power consumption in bounded latency. Aiming to connect multi-hop lossy 802.15.4e TSCH networks into the Internet, IETF created the 6TiSCH working group for establishing interoperability between TSCH and IPv6. This working group currently working on IPv6-enabled communication protocol stack in which one task of the MAC layer is the network formation where new nodes join into the existing network.

The 6TiSCH defines minimal configuration [1] standard which specifies that all nodes should transmit their bootstrapping traffic and other control messages in shared slots. A node, intend to join with the existing network, needs to contend using random channel access protocol such as carrier sense multiple access with collision avoidance (CSMA/CA) to transmit their packets in a shared slot. The minimal configuration defines only one shared slot in each slotframe or multi-slotframe. A slotframe constitutes of several timeslots, and one timeslot length is defined to be long enough to transmit a data frame and receive an acknowledgement (ACK) frame. During network formation, several control frames are transmitted. For example, Enhance Beacon (EB) frames are sent by already joined nodes, join request (JR) frames are sent by the new joining nodes, and RPL [2] DIO packets are sent by already joined node to construct destination oriented directed acyclic graph (DODAG) topology. Therefore, the rate of transmitting these frames/packets in a limited number of shared slots plays a significant role in network formation process of 6TiSCH network.

Existing works on network formation [3]–[7] considered that only EB frame is sent during the network formation process. Again, in all the existing works, the authors allocate more bandwidth i.e. more number of shared slots for transmitting EB packets once the network has many joining nodes. In a recent work, Vallati *et al.* [8] mentioned that a new node should receive both EB and DIO packets to join the network during the network formation. They also analytically proved that the minimal configuration does not provide enough resources to send these control frames. To overcome the problem of unavailability of sufficient resources i.e. shared slots, the authors allocate more number of shared slots to transmit more control frames/packets in less time. However, the allocation of extra shared slots costs in higher energy consumption. Further, it does not follow the minimal configuration standard. Vucinic *et al.* [9] considered that along with the EB and DIO, JR frames are also transmitted during network formation. Consequently, they proposed that the beacon transmission probability should be 0.1 irrespective of the number of nodes present in the network for faster network formation. However, we observe that the performance of a 6TiSCH network degrades when a new node joins the network. Therefore, we propose a dynamic beacon interval allocation scheme in which the interval varies with the number of joined nodes during network formation.

We summarize the main contributions of this paper as

978-1-5386-7902-9/19/$31.00

follows:

- Analytically we prove that fixed beacon interval (BI) in minimal configuration degrades the network performance with respect to joining time once a new node joins.
- We propose a dynamic beacon interval scheme for faster association of nodes in 6TiSCH network.
- Finally, we validate our proposed approach using simulation.

II. Network Formation Analysis

In a 6TiSCH network, initially a personal area network (PAN) coordinator periodically broadcasts EB frames, which contains all the necessary information for a node to join the network. A new node randomly scans on all the available channels (at most 16 channels according to minimal configuration) to receive an EB. Once it receives an EB, it sends JR frame to PAN coordinator or to the already joined node from which he receives the EB frame. A node is eligible to send his own beacon, for further expansion of the network, once it receives the acknowledgement (ACK) of its JR frame transmission followed by the reception of atleast one DIO packet.

Symbol	Meaning
N	Number of joined nodes
K	Number of new nodes
N_c	Number of available channels
L	Slotframe length
I_{eb}	Beacon generation interval
P_{eb}	Transmission probability of EB in a shared slot
P_{dio}	Transmission probability of DIO in a shared slot
P_{jr}	Transmission probability of JR in a shared slot
P_{loss}	Packet loss probability
P_{EB_S}	Probability of successfully transmitting an EB frame
P_{DIO_S}	Probability of successfully transmitting a DIO packet
P_{JR_S}	Probability of successfully transmitting a JR frame

TABLE I: List of used symbols

At the outset, Table I summarizes all the used symbols and their corresponding meaning. Let us consider a scenario in which N number of nodes are already joined in the network and K new nodes are trying to join the network. Assume that all nodes are neighbour to each other. Let P_{eb}, P_{dio}, P_{jr} are the transmission probability of EB frame, DIO packets and JR frames, respectively. P_{loss} is the packet loss probability and N_c is the total number of channels used in the network. Then, the probability of sending an EB frame successfully in a shared slot is computed as,

$$P_{EB_S} = \frac{NP_{eb}(1-P_{eb})^{N-1}(1-P_{dio})^{N-1}(1-P_{jr})^{K}(1-P_{loss})}{N_c} \tag{1}$$

This is because, when a node sends its own EB frame, the remaining $(N-1)$ joined nodes neither should send their EB frames nor any DIO packet. The probability of this step equals $(P_{eb}(1-P_{eb})^{N-1}(1-P_{dio})^{N-1})$. Additionally, the new nodes also should not send their JR frames which is defined by the probability equals $((1-P_{jr})^K)$. The transmitted frame should not be lost in the channel with probability equals $(1-P_{loss})$. Initially, a new node is not synchronize with its coordinator's channel hoping sequence. Therefore, it searches in all the N_c number of channels which reduces the EB success probability by N_c times.

We know that a node can send JR frames only when it successfully receives atleast one EB frame. Therefore, $P_{jr} = P_{EB_S}$. So, we rewrite the Equation (1) as,

$$\frac{P_{EB_S}}{(1-P_{EB_S})^K} = \frac{NP_{eb}(1-P_{eb})^{N-1}(1-P_{dio})^{N-1}(1-P_{loss})}{N_c} \tag{2}$$

Again, $|P_{EB_S}| < 1$. Thus, using binomial approximation, we can write

$$(1-P_{EB_S})^K = 1 - KP_{EB_S}$$

Now, the Equation (2) can be rewritten as,

$$\frac{P_{EB_S}}{(1-KP_{EB_S})} = \frac{NP_{eb}(1-P_{eb})^{N-1}(1-P_{dio})^{N-1}(1-P_{loss})}{N_c} \tag{3}$$

Let us consider $\frac{NP_{eb}(1-P_{eb})^{N-1}(1-P_{dio})^{N-1}(1-P_{loss})}{N_c} = X$. Then, from Equation (3), we get,

$$P_{EB_S} = \frac{X}{1+KX} \tag{4}$$

Similarly, considering that EB frame has strict priority over DIO packets, the probability of sending a DIO packet successfully in a shared slot is computed as,

$$P_{DIO_S} = NP_{dio}(1-P_{eb})^N(1-P_{dio})^{N-1}(1-P_{EB_S})^K(1-P_{loss}) \tag{5}$$

Note that, in this equation, N_c is not used because a node knows its coordinator's channel hopping sequence after getting an EB frame. Likewise, the probability of sending a JR frame successfully in a shared slot is also computed as,

$$P_{JR_S} = KP_{EB_S}(1-P_{EB_S})^K(1-P_{eb})^N(1-P_{dio})^N(1-P_{loss}) \tag{6}$$

Now, we should know the transmission probability of EB, DIO and JR in a shared slot. The probability P_{eb} can be calculated as,

$$P_{eb} = \frac{L}{I_{eb}} \tag{7}$$

where, L is the slotframe length and I_{eb} is the beacon interval. We compute the probability P_{dio} following the procedure describe in [8], and considering the similar type of assumption it can be directly written as,

$$P_{dio} = (1-P_{eb})\frac{2^{N_D}(1-P_r)^{N_D}min(\frac{L}{2^{N_D}I_{min}},1) + \sum_{i=0}^{N_D-1}P_r2^i(1-P_r)^i min(\frac{L}{2^iI_{min}},1)}{P_r + 2^{N_D}(1-P_r)^{N_D} + \sum_{j=1}^{N_D-1}P_r2^j(1-P_r)^j} \tag{8}$$

Here, I_{min} is the intial DIO interval, P_r is the trickle algorithm reset probability and N_D represents number of trickle algorithm states.

Thus, we can compute the average joining time of a new node using the Equations (4), (5) and (6) as follows,

$$AvgJoinTime = (\frac{1}{P_{EB_S}} + \frac{1}{P_{DIO_S}} + \frac{1}{P_{JR_S}})L \quad (9)$$

Finally, we analytically asses the performance of the minimal configuration standard. Using the above mentioned scenario and assigning values to the variables such as $N_C = 16$, $I_{eb} = 2 \times L$, $P_{loss} = 0.2$, $I_{min} = 8ms$, $L = 5$ timeslots, timeslot duration = 10ms, $N_D = 16$, $P_r = 0.2$, we draw the Fig. 1 in which the average the joining time is computed by the Equation (9). The figure clearly depicts that, for a fixed beacon interval, the joining time of nodes increases with the increase in the number of already joined nodes. In other words, the performance of the network degrades once it allows to join a new node with it.

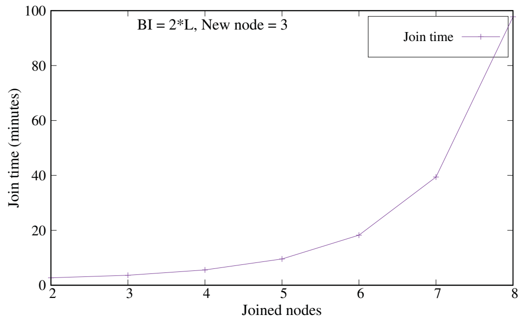

Fig. 1: Number of nodes Vs Join time

We have varied the beacon interval randomly to asses its impact on joining time for a fixed number of joined nodes and new nodes. The received result is plotted in Figure 2. This figure also clearly shows that joining time of a node varies with the beacon interval for a fixed number of nodes.

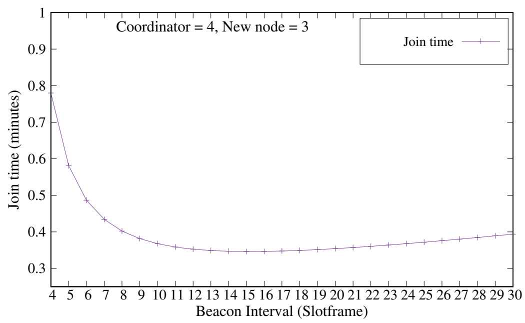

Fig. 2: Beacon interval Vs Join time

Apart from this analytical model, we have also analysed the minimal configuration using simulation in Cooja tool on Contiki OS. The simulation parameters and their corresponding values are mentioned in Table II. In this simulation, instead of taking a random topology, we took a kite like topology as shown in Figure 3, where Node-4 is under the coverage area of four other nodes and used for network expansion. The joining time of the last (i.e., Node-7) is plotted in Figure 4. This graph also shows that the overall network formation time is highly dependant on the beacon interval used by the nodes in the network. Therefore, from these figures, we can conclude that faster joining time can be achieved by tuning beacon interval.

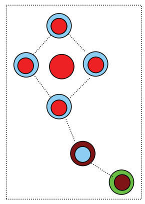

Fig. 3: Kite Topology

Parameter	Value
Timeslot length	10 ms
Slotframe	5 timeslots
Number of channels	16
RPL interval	Trickle Algo.
Mote type	Cooja mote

TABLE II: Simulation parameters

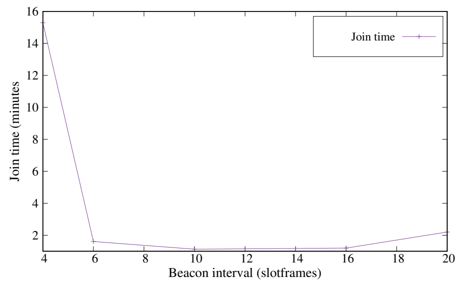

Fig. 4: Simulation results for Kite Topology

III. Methodology

In this paper, we propose a scheme to dynamically adjust the beacon interval depending upon the congestion in shared slots, which indirectly depends on the number of joined nodes. Congestion estimation can be done using different methods considering many parameters such as number of neighboring nodes, packet loss, queue length, and channel load. In this work, we use channel busyness ratio (CBR) [10] for congestion estimation as it gives better result in both the static and dynamic networks. We measure CBR for a particular period of time and then we allocate beacon interval to a node based on the current CBR. The assigned beacon interval will be used in the next time period.

The CBR for a particular period of time is calculated as,

$$CBR = \frac{Busy\ shared\ slots}{Busy\ shared\ slots + Empty\ shared\ slots} \quad (10)$$

A shared slot is considered to be busy if the measured signal strength is greater than the clear channel assessment (CCA). Otherwise, a shared slot is considered as an empty slot. After computing the CBR for a particular time interval, a node computes its own beacon interval for the next time period as follows,

$$I_{eb} = Max\ (Min\ I_{eb},\ (Max\ I_{eb})^{\ CBR}) \quad (11)$$

Using the above equation, a node will always choose its beacon interval in between the minimum beacon interval $Min\ I_{eb}$ and maximum beacon interval $Max\ I_{eb}$ as the values of CBR will remain within the range from 0 to 1.

IV. Performance Evaluation

For the performance evaluation of the proposed scheme, we randomly deployed N nodes in a fixed square size grid where the value of N varies from 5 to 25. We used the simulation parameters and network environment as used for the kite topology, in Section II, except the slotframe length which equals 10. In minimal configuration, we took beacon interval equals $16\times$slotframe length. In the proposed scheme, we took the same value for minimum beacon interval and we consider its maximum value equal to $32\times$slotframe. For CBR calculation, we use the time period interval equals $32\times$slotframe.

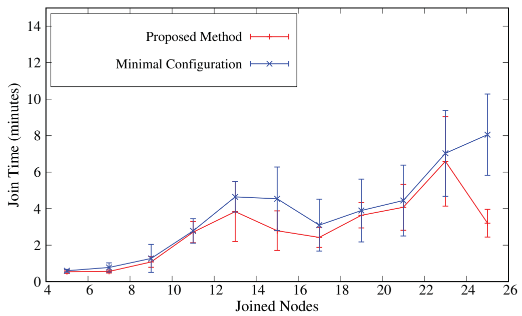

Fig. 5: Nodes joining time in random topology

Figure 5 depicts the variations in node joining time with respect to the number of joined nodes. We plot the result with 95% confidence interval. This graph shows that the proposed approach outperforms the minimal configuration in terms of joining time. This is because when the number of nodes increases in the network, due to the fixed beacon interval in minimal configuration, all joined nodes generate EB frame in same rate which results in congestion in the shared slots. Hence, a new node takes longer time to join the network as it needs to successfully receive and send packet/frame for association. However, in the proposed approach, beacon generation rate decreases (i.e. generation interval increases) with the increase of the number of joined nodes resulting less congestion in the shared slots, and thus, lesser joining time.

V. Conclusion

In this paper, a dynamic beacon interval scheme for faster association of nodes in 6TiSCH networks is introduced. At first, we show, using both analytical model and simulation, that due to fixed beacon interval in minimal configuration the performance of a network degrades with the increase of number of joined nodes. To overcome this problem, we proposed a dynamic beacon interval scheme in which we assigned a beacon interval to a node depending on its present CBR value. The proposed scheme is evaluated using simulation, and the received result is further compared with the benchmark protocol – minimal configuration standard. Simulation results prove that the proposed scheme significantly reduces overall joining time.

References

[1] X. Vilajosana, K. Pister, and T. Watteyne, "Minimal IPv6 over the TSCH Mode of IEEE 802.15.4e (6TiSCH) Configuration," RFC 8180, May 2017.

[2] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. P. Vasseur, and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," IETF RFC 6550, Mar. 2012.

[3] E. Vogli, G. Ribezzo, L. A. Grieco, and G. Boggia, "Fast join and synchronization schema in the IEEE 802.15.4e MAC," in *Proceedings of IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, March 2015, pp. 85–90.

[4] D. D. Guglielmo, A. Seghetti, G. Anastasi, and M. Conti, "A performance analysis of the network formation process in IEEE 802.15.4e TSCH wireless sensor/actuator networks," in *Proceedings of IEEE Symposium on Computers and Communications (ISCC)*, June 2014, pp. 1–6.

[5] E. Vogli, G. Ribezzo, L. A. Grieco, and G. Boggia, "Fast network joining algorithms in industrial IEEE 802.15.4 deployments," *Ad Hoc Networks*, vol. 69, pp. 65–75, 2018.

[6] D. D. Guglielmo, S. Brienza, and G. Anastasi, "A model-based beacon scheduling algorithm for IEEE 802.15.4e TSCH networks," in *Proceedings of IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2016, pp. 1–9.

[7] T. P. Duy, T. Dinh, and Y. Kim, "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*, vol. 12, no. 8, 2016.

[8] C. Vallati, S. Brienza, G. Anastasi, and S. K. Das, "Improving network formation in 6TiSCH networks," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2018.

[9] M. Vucinic, T. Watteyne, and X. Vilajosana, "Broadcasting Strategies in 6TiSCH Networks," *Internet Technology Letters*, Dec. 2017. [Online]. Available: https://hal.inria.fr/hal-01630316

[10] M. Khatua and S. Misra, "D2D: Delay-aware distributed dynamic adaptation of contention window in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 2, pp. 322–335, Feb 2016.