Channel BlaQLisT: Channel Blacklist using Q-Learning for TSCH

JunMyeung Kim, Sang-Hwa Chung

Department of Information Convergence Engineering Pusan National University

Busan, South Korea

chrismail@naver.com, shchung@pusan.ac.kr

***Abstract*—** In time-slotted channel hopping (TSCH) communications included in IEEE 802.15.4, there are no direct scheduling guidelines, but communication schedules are determined by the distribution of slots according to Slotframes and the allocation of channels. Among them, channel allocation is directly exposed to effects such as frequency mix caused by the increasing number of wireless communication equipment. In this paper, we propose a technique to create a Hopping sequence list based on the Blacklist according to the channel quality of each slot by directly matching the Q-table of Q-learning, a technique of reinforcement learning, to maintain the quality and maximize the efficiency of communication. To verify the above technique, Contiki-NG's Cooja simulation was verified, and one of the existing studies was selected as a comparison group. Simulation results show higher PDR and energy charge compared to previous studies, showing relatively good efficiency, reliability, and the potential of this technique.

Keywords—IEEE 802.15.4, TSCH, Channel blacklisting, Reinforcement learning, Q-learning, Q-table, Channel quality prediction, PDR(Packet Delivery Ratio), WSN

I. Introduction

The rise of radio equipment used in modern society grows the frequency density in wireless environments, causing an increase in "Cross Technology Interference (CTI)"[1]. The time-slotted channel hopping (TSCH), the MAC layer communication of the IEEE 802.15.4 standard [2], is designed to avoid interference by using the division and distribution of channels and slots on its own (“TSCH makes use of pseudorandom channel hopping to combat external interference and frequency-selective multipath fading.” [3]). Since there is no consideration of the surrounding environment, it is a method that is not free from the influence of “CTI” [1]. In this paper, we propose a channel blacklist method to efficiently minimize the effect on “CTI” [1] of frequency.

This idea applies Q-table of Q-learning [4] to improve communication quality degraded by interference. After matching the Q-table to the TSCH Schedule table consisting of Slotframe and channel, the channel quality for each slot is evaluated using the PDR evaluated for each communication as a metric, and channels that do not have a certain level are dropped and the best channel is brought. Adaptive Channel Selection [3], which implemented channel hopping using RSSI as a metric, was selected as a verification method for this technique. Comparing the technology with PAR, PDR, RPL parent switching counts, and energy charge of the Cooja simulator in the Contiki-NG OS environment, it was found that overall, much better reliability and efficiency were achieved.

II. Related Works

There are many studies conducted on channel hopping to improve the TSCH technique of IEEE 802.15.4. MABO-TSCH [5], which learns PDR-based channel selection with MAB (Multi-Armed Bandit) problem; Adaptive Channel Selection(ACS)[3], which selects a channel based on the RSSI value of every communication; ATSCH [6] which selects “ED (Energy Detection)” [6] using an empty slot and the energy of each channel during one Slotframe; ETSCH [7], which selects a channel with “NICE (Non-Intrusive Channel Quality Estimation)” [7], that is a method in which ED is implanted in every slot rather than allocating an ED to an empty slot by improving the ATSCH [7]; ITSCH[1], which learns the frequency energy level obtained in advance using the NICE technique taken from ETSCH with DNN (Deep Neural Network) and applies it to channel selection; Adaptive Channel Capacity Shaping(ACCS), a per-link solution that selectively adjusts the number of transmission opportunities based on physical channel quality[10]; a method of calculating the time-varying characteristics of external interference using dynamic MABB(Bernoulli compensation/penalty MAB) method[11, 12]; etc. In the case of BlaQList to be described in this paper, the algorithm is changed to Q-learning in MABO-TSCH using MAB problems and the channel quality evaluation applied in one Slotframe unit is improved in other studies to be applied in each slot unit for precise operation.

A. TSCH(Time-Slotted Channel Hopping)

The TSCH technique introduced in IEEE 802.15.4[2] is a MAC layer communication technique that improves the reliability of communication and is intended for use in the IoT domain through a combination of a Slotframe in the time domain and a channel in the frequency domain. The channel consists of up to 16 channels, and in the case of Slots, it can be declared and used as necessary, but as the total Slotframe length increases, the communication schedule may have a longer term. In this paper, we design to use 16 channels and 31 slots. ("Figure 1", allocating 10 ms per slot, the Slotframe is 310 ms.) The default TSCH can be channel (CH) identified by (1) based on Absolute Sequence Number (ASN) and a predetermined Channel Offset and can now be slot estimated based on Slotframe Length. HSL stands for TSCH's Hopping Sequence List. This list is the best channel list that will be changed according to the Blacklist reflecting the channel quality.

$$Slot = [\,(ASN) mod (SlotFrame\ length)\,]$$
$$CH = HSL\,[\,(ASN + CH_{Off}\,)\, mod \mid HSL \mid\,] \qquad (1)$$

B. Q-Learning

Q-learning, a type of reinforcement learning, is a method of reinforcing actions corresponding to actions and states by identifying the state derived from the action performed by the agent between the environment and the agent and updating the Q-table with reward. ("Fig. 2", in this paper, Environment=slot, Agent=node, Action=Channel Select.) The basic form of Q-table is "Fig. 3" and the basic formula is (2).

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_{t+1} + \gamma \max_{a_{t+1} \in A(s_{t+1})} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t))$$
$$\alpha: learning\ rate, 0 < \alpha < 1$$
$$\gamma: discount\ factor, 0 < \gamma < 1 \qquad (2)$$

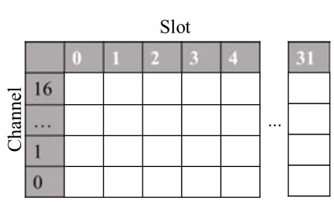

Fig. 1 Time-Slotted Channel Hopping, TSCH Schedule table

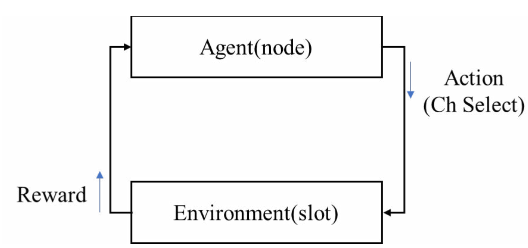

Fig. 2 Reinforcement Learning

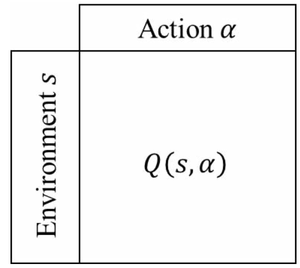

Fig. 3 Basic Q-table

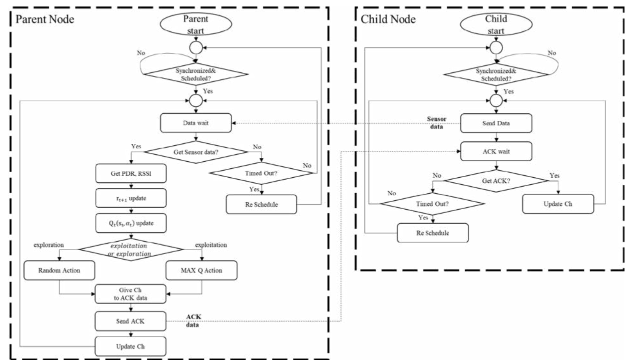

Fig. 4 Flow Chart

III. Proposed Algorithm

For TSCH Network Quality and Reliability, this paper proposes a Blacklist based on reinforcement learning. This algorithm consists of three stages, and HSL is distributed from coordinator to child node after each Reward calculation, Q-table update, and Blacklist sharing. "Fig. 4" is Flow Chart for this algorithm. Briefly explaining the Flow chart, the sensor data collected from the child node that has completed synchronization is transmitted. When sensor data is received from the parent node, compensation r is calculated based on PDR and RSSI and Q is updated. After that, according to the ε-greedy strategy, a channel according to Random or Max Q is selected, and the updated value of the blacklist is included in the ACK and then sent. Then, the channel is applied by the Parent Node itself. In case of failure to receive Sensor Data by the Parent Node, rescheduling is initiated in case of Time Out, otherwise, it continuously waits for the reception. When ACK is received by the child node, the channel is updated with the newly received blacklist and the sensor communication begins again. If the child node fails to receive the ACK, rescheduling is performed if it is Time Out, and if not, it continuously waits for the reception.

A. Reward Calculation

Reward is an important value for fitting in reinforcement learning, and a certain and accurate value must be selected. In the case of this paper, since the communication model is targeted, it was decided that it is correct to use the communication success/failure as an indicator of learning. If the communication success is quantified, the Packet Delivery Ratio (PDR) is indeed a quantitative representation, so it has been specified as an indicator value for the reward. By default, the value of reward starts at 90. This figure is determined because good communication basically guarantees a communication success rate of more than 90%. Reward formula consisting of the above premise (3).

$$r_{t+1} = PDR \times r_t, \qquad r_0 = 90 \tag{3}$$

B. Update Q-table

The basic Q value equation (2) has a discount factor, so it fixes the Q-table after a period and fixes the performance after learning is finished. It is considered zero and designed to continuously respond. The learning rate was given 0.2 which is commonly used. In Equation (4), the Q-table derived from Slotframe and Channel is the Blacklist "Fig. 5" to be used in this paper.

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_{t+1} - Q(s_t, a_t)) \tag{4}$$

C. Blacklist Advertising

Since the communication bandwidth is too large to use the Blacklist "Fig. 5", created by the Coordinator for Downstream. So only the channel quality list corresponding to the current slot is extracted through (1), and the channel corresponding to the arbitrarily set PDR 95% Threshold is applied to HSL and advertised. Evaluation to verify the BlaQLisT technique, an experiment was conducted using the Cooja simulator [8] of Contiki-NG, the latest version of Contiki OS.

D. Experiment

In this experiment, to exclude performance change through slot hopping and induce channel contention, the limit was applied to one slot shared for Sync and one unicast for data communication. The experiment proceeds to check the performance of each of the three topology cases. First, Square Grid Type where each node is connected one-to-one; Second, Ellipse Type, in which nodes are symmetrically arranged and composed of multi-hop; Third, Asymmetry Grid Type, which is asymmetrically arranged, and multi-hop and multi-path are created. There are four evaluation factors as performance indicators. The number of RPL switches to check PAR (Packet Acknowledgment Ratio) and link quality. Charge current for end-to-end PDR (Packet Delivery Ratio) energy consumption measurement to verify final communication quality. The experimental conditions are as follows.

Environment Current Slot	Action Channel Select						
	0	**1**	**2**	**3**	**4**		**31**
16							
…						…	
1							
0							

Fig. 5 Blacklisted Q-table (BlaQLisT)

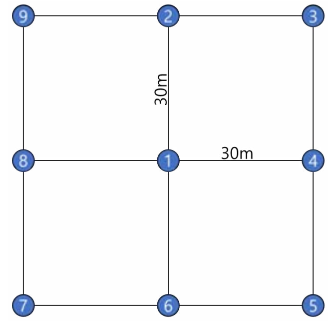

Fig. 6 Square Grid Type Topology

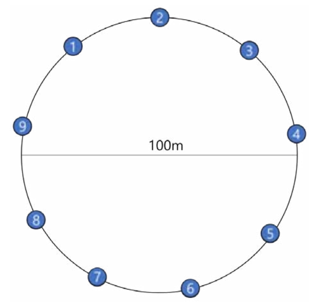

Fig. 7 Ellipse Type Topology

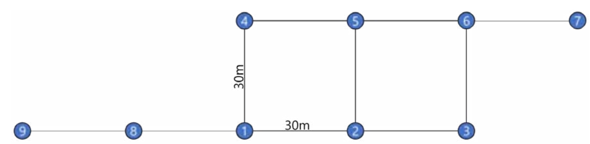

Fig. 8 Asymmetry Grid Type Topology

- Simulator: Contiki-NG Cooja
- Node (Radio CPU cc2650): 9 EA
- Channel: 16 ch
- Slotframe: 31 slots
- Simulation Time: 1 hour
- Routing: RPL [9]
- Topology: "Fig. 6", "Fig. 7", "Fig. 8"
- Using slot: 2 slot (1 shared, 1 unicast)

E. PAR(Packet Acknowledgement Ratio)

It is calculated by comparing the Acknowledgment Counts and Tx Counts used to maintain the communication link (5). According to the experiment, the performance was improved by 6.27%, 35.56%, and 36.78%, respectively"Fig. 9". In terms of link quality, it can be said that the performance is at least similar to ACS and the maximum result shows considerable performance improvement.

$$PAR = (100 \times parent\ packet\ Ack) \div parent\ packet\ tx \tag{5}$$

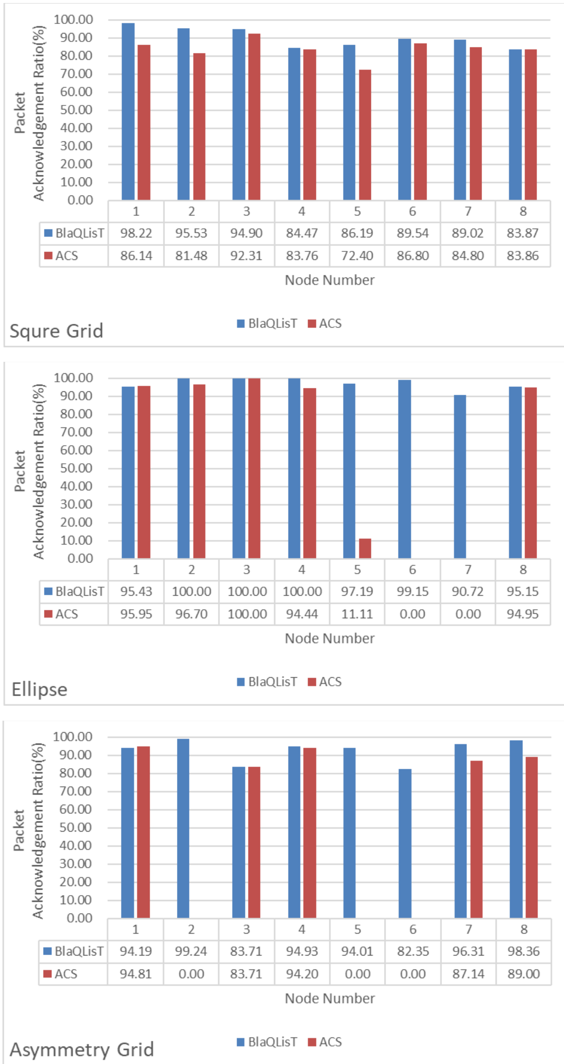

Fig. 9 PAR Charts

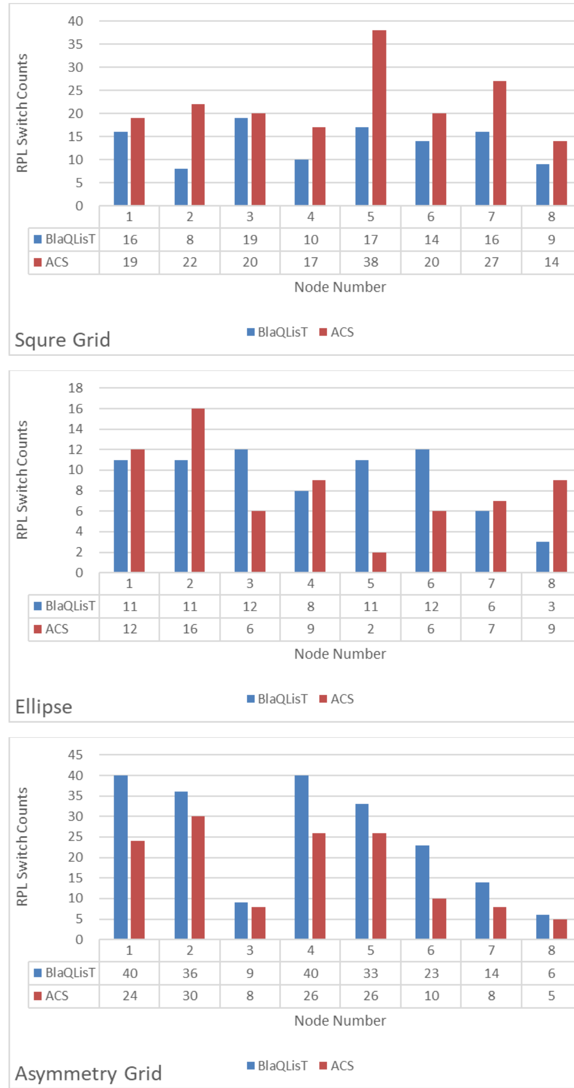

Fig. 10 RPL Switch Charts

F. RPL Switch Counts

RPL [9] is a method of constructing a route by forming a DoDAG as a communication routing technique. In this paper, we verify how consistent the DoDAG of Standard RPL Routing is to verify that it is optimized for the communication environment. That is, the number of RPL Parent ID changes is measured and the smaller the number, the better the performance is judged. As a result of the experiment, RPL Parent fluctuated 8.5 times less, 0.8 more times, and 8 more times, respectively “Fig. 10”. However, to reflect the overall result, as shown in “Fig. 9”, there is a node with a problem in the formation of RPL itself, making it difficult to use the ACS side result as it is. Accordingly, if we consider only the Square Grid Type, which has no problem in RPL formation, it can be judged that the performance has been improved.

G. PDR(Packet Delivery Ratio)

PDR is the most important indicator in this experiment and refers to the data packet transmission success rate and is defined as Data Rx Count compared to Data Sequence Number (6). Numerically, this indicator shows a big difference with an average improvement of 3.33%, 8.34%, and 22.58%, respectively. In addition, if you look at the difference in PDR between the minimum and maximum values in "Fig. 11", it is judged that the quality retention is excellent, even apart from the overall average quality.

$$PDR = (100 \times packet\ Delivered) \div Total\ packet \quad (6)$$

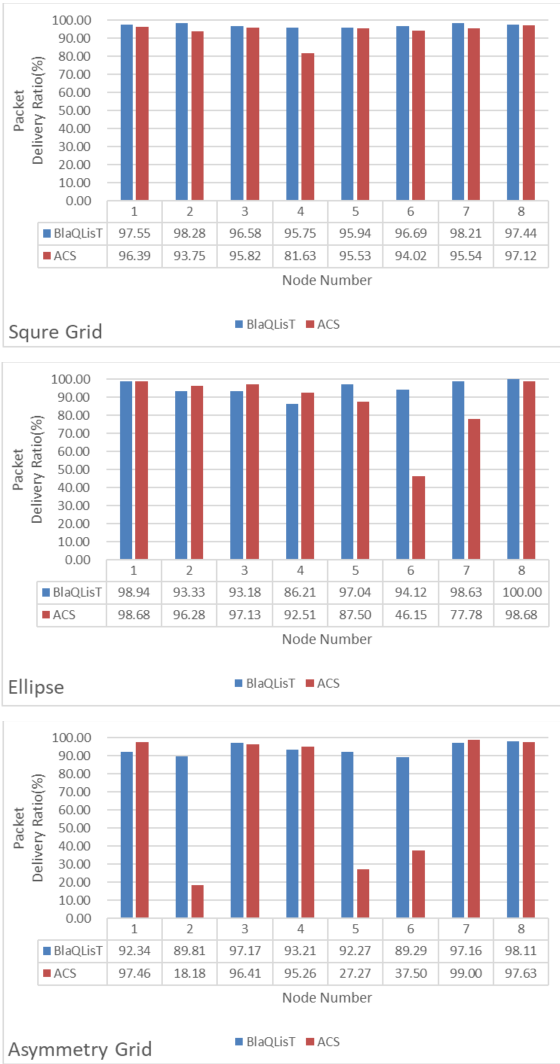

Fig. 11 PDR Charts

Fig. 12 Current Charge Charts

H. *Current Charge*

In Cooja Simulator, the estimated current consumption for the CC2650 is defined. By multiplying and summing the time per sequence and the estimated current consumption, it is possible to estimate the total current for each node (7). We improved energy consumption by 8.7%, 6% and 39.6% respectively “Fig. 12”.

$$\begin{aligned} Current\ Charge = \ & (T_x\,Time \times T_x\,Current) \\ & +(R_x\,Time \times R_x\,Current) \\ & +(CPU_{on}\,Time \times CPU_{on}\,Current) \\ & +(CPU_{sleep}\,Time \times CPU_{sleep}\,Current) \\ & +(CPU_{deep}\,Time \times CPU_{deep}\,Current) \end{aligned} \tag{7}$$

$T_x\,Current = 9.1\,mA$,
$R_x\,Current = 5.9\,mA$,
$CPU_{on}\,Current = 0.061\,mA$,
$CPU_{sleep}\,Current = 1.335mA$,
$CPU_{deep}\,Current = 0.01\,mA$

IV. CONCLUSION

In this work, we propose a new type of frequency channel hopping using Q-learning, a form of reinforcement learning. The Q-table of Q-learning is matched with TSCH's Schedule table, which consists of Slotframes and channels, so unlike many existing methods, it is designed to train channels for each slot to calculate more fitted results. It is simple and effective because it implements fitting only with the result of communication (PDR) using the principle of observing the result, which is the basic characteristic of reinforcement learning, and training accordingly. Compared to the Adaptive Channel Selection technique using RSSI metric, the average performance improvement of PAR 6~36% and PDR 3~22% were improved, although it is limited to normal cases, the number of RPL switches was reduced by 6%, and consistent PDR distribution was verified to prove more stable communication. In addition, compared to the comparison system, energy consumption was improved, and the potential was confirmed in terms of economic feasibility by reducing close to 40% under a specific topology.

In future research, we plan to add Q-table scale algorithms to be used in actual testbeds, design them to adapt to changes such as Slotframes, and apply DQN, an advanced form of reinforcement learning, to functions.

ACKNOWLEDGMENT

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2022-2016-0-00318) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation)

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-01450, Artificial Intelligence Convergence Research Center [Pusan National University])

REFERENCES

[1] Mohammad Farahmand, Majid Nabi, "Channel Quality Prediction for TSCH Blacklisting in Highly Dynamic Networks: A Self-Supervised Deep Learning Approach" IEEE Sensors Journal, vol. 21, Issue. 18, Sept 2021. IEEE.

[2] "IEEE Standard for Local and metropolitan area networks—Part 15.4," IEEE Std 802.15.42015, 2015.

[3] Atis Elsts, Xenofon Fafoutis, Robert Piechocki, and Ian Craddock, "Adaptive Channel Selection in IEEE 802.15,4 TSCH Networks" 2017 Global Internet of Things Summit(GIoTS), August 2017. IEEE.

[4] Christopher J.C.H. Watkins, Peter Dayan, "Q-learining" Machin learning 8, pp. 279-292, May 1992. Kluwer Academic.

[5] Pedro Henrique Gomes, Thomas, Watteyne, Bhaskar Krishnamachari, "MABO-TSCH:Multihop and blacklist-based optimized time synchronized channel hopping" Transactions on Emerging telecommunications technologies, Vol. 29, Issue. 7, e3223, August 2017. WILEY.

[6] Peng Du, Dr. George Roussos, "Adaptive Time Slotted Channel Hopping for Wireless Sensor Networks" 2012 4th Computer Science and Electronic Engineering Conference(CEEC), pp. 29-34, Sept 2012. IEEE.

[7] Rasool Tavakoli, Majid Nabi, Twan Basten, Kees Goossens, "Enhanced Time-Slotted Channel Hopping in WSNs using Non-Intrusive Channel-Quality Estimation" 2015 IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems, pp. 217-225, Oct 2015. IEEE.

[8] Fredrik Österlind, Adam Dunkels, Joakim Eriksson, Niclas Finne, Thiemo Voigt, "Cross-Level Sensor Network Simulation with COOJA" Proceedings. 2006 31st Conference on Local Computer Networks, pp. 641-648, Nov 2006. IEEE.

[9] T. Winter et al., "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks" RFC 6550, March 2012. IETF

[10] Gianluca Cena, Stefano Scanzio and Adriano Valenzano, "Ultra-Low Power Wireless Sensor Networks Based on Time Slotted Channel Hopping with Probabilistic Blacklisting", Electronics 2022, Jan 2022. MDPI

[11] Nastooh Taheri Javan, Masoud Sabaei , and Vesal Hakami, "Adaptive Channel Hopping for IEEE 802.15.4 TSCH-Based Networks: A Dynamic Bernoulli Bandit Approach", IEEE Sensors, Vol. 21, Issue. 20, Oct 2021. IEEE

[12] X. Lu, N. Adams, and N. Kantas, "On adaptive estimation for dynamic Bernoulli bandits," Found. Data Sci., vol. 1, no. 2, pp. 197–225, 2019