Vol.2018-MBL-89 No.15
Vol.2018-ITS-75 No.15
2018/11/16

Feasibility Study on Distributed Sensor Processing in BLE Mesh Network

Stirapongsasuti Sopicha[1] Nakamura Yugo[1] Suwa Hirohiko[1] Arakawa Yutaka[1,2]
Yasumoto Keiichi[1]

Abstract: Machine Learning (ML) based on in-network processing in a wireless sensor network (WSN) is attracting attention. However, energy consumption is a critical issue to be solved. Aiming to explore the feasibility of deep learning over WSN, we have been developing a low power sensor node consisting of Intel Edison for processing and SenStick for sensing with an energy harvesting module by solar panel. In our previous system, SenStick can operate with energy harvesting but Edison still consumes a lot of power for processing and networking. In this paper, aiming to achieve much lower power consumption for ML on WSN, we investigate a feasibility of using an embedded IoT MCU for both processing and networking. For this purpose, we develop a simple application on top of BLE mesh network formed by more than 5 MCU boards with sensors and investigate the power consumption and processing performance of such embedded system based on BLE mesh network.

Keywords: Wireless Sensor Network, BLE Mesh Network

1. Introduction

In the past decade, wireless sensor network (WSN) has seen a renewed importance in indoor services. As the sensor network can support to monitor and control various types of appliances in buildings, the possibility of various CPS (cyber physical systems) services such as sensing environment temperature of different places and controlling the air condition temperature should be explored to advocate for a community. In order to understand what happens in the environment, analysis of sensor data sensed at different places in the target area is essential. Machine learning (ML) at cloud servers is a major approach for this purpose, but it wastes wide-area wireless network resource from sensors to cloud servers. Therefore, realizing ML paradigms on WSN is attracting increasing attention.

Recently, some ML methods on WSN have been proposed. For example, Taherkordi et al.[1] presented the method to reduce data transmission, processing time and power consumption of sensor nodes with the distributed k-Means clustering algorithm to group data, summarize them and send towards a sink node. Li et al.[2] leveraged Deep Neural Network (DNN) to construct a simulated WSN by dividing DNN layers and deploying them into WSN nodes. The purpose is to decrease power consumption in WSN by reducing the number of transmitted data processed on DNN at each node. Since our research focuses on the practical application, selecting the suitable hardware as a node in WSN should be considered. Also, Fukushima at al. [3] proposed a new paradigm of attributed CNN learning on the top of mesh sensor network, namely Microdeep which achieved a comparable performance to ordinary CNNs.

In the previous research [4], we developed a low power sensor node using Intel Edison and an ultra small multi-sensor called SenStick [4]. Also, we included an energy harvesting module for self-generated power with a solar panel. However, the characteristic of the node has not been dealt with in depth due to an inefficient power consumption of Intel Edison on processing and networking.

Thus, we have developed the new sensor node and changed an embedded IoT hardware from Intel Edison to Cypress Pioneer Baseboard Preloaded with CY8CKIT-142 PSoC4 BLE Module (PSoC4 BLE). The PSoC4 BLE is a 32-bit MCU. It uses ARM Cortex-M0 CPU which operates at very low power from 1.71 to 5.5V compared to Intel Edison using voltage ranged from 7V to 15V.

Additionally, in this research, we implement the mesh network topology for WSN because this topology offers connectivity between any nodes in WSN, so it can encourage ML to realize application services on the top of the mesh network. Although the mesh network topology is interesting, there is a challenge in developing mesh network on BLE modules since a BLE module can perform only one role at one time as central or peripheral, leading to difficulty in multi-hop communication. We solve this problem by changing roles at each sensor node depending on time slots so that neighboring nodes form central-peripheral pairs at each time slot and can communicate with each other.

This paper investigates the feasibility of BLE mesh network when all nodes in the network connect to their environment sensors and attempt to update their sensor data though mesh network. To evaluate the performance of mesh network, we set up the

1 Graduate School of Science and Technology, Nara Institute of Science and Technology, Ikoma, Nara 630–0192, Japan
2 JST PRESTO

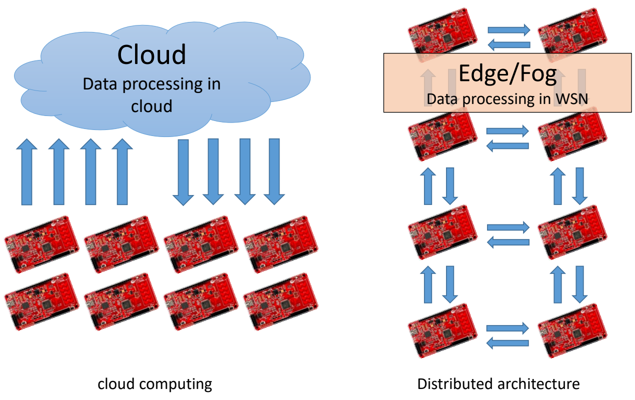

Fig. 1 Data processing architecture

test-bed experiments starting with a simple BLE mesh network. The experiment measured the data transmission time, power consumption and packet loss rate, then comparing results of each network at different number of nodes.

The following sections explain related work, an approach to construct the BLE mesh network, a test-bed experiment, results and conclusion, respectively.

2. Related work

2.1 Distributed data processing

Cisco predicted that 50 billion things would be connected to the Internet by 2020 [5]. IoT devices will come into popular use in various environments such as office, school, shopping mall, and so on. On the other hand, new issues are caused such as a dramatic increase in data traffic and power consumption by those IoT devices, because most of the IoT-based services are based on the centralized architecture at cloud servers.

In cloud architecture, each IoT device uploads data to the cloud server and data is processed on the cloud and finally, the result is delivered to the user. Accordingly, not only does the user suffer from large delays to get the service but also the data traffic to the cloud as well as the power consumption of the cloud is quickly growing.

To solve this problem, distributed edge/fog computing architectures have been proposed such as IFoT[6], [8] and DIAT[7]. Fig. 1 shows the difference between cloud computing and the distributed edge/fog computing architectures. The distributed edge/fog computing architecture needs processing at sensor/IoT nodes which are typically driven by a battery. How to reduce the power consumption at sensor nodes is an emerging issue to be solved. And also, since data is processed in the wireless sensor network (WSN), it is important to develop a mesh network.

2.2 Wireless mesh network

Mesh network has been widely addressed in WSN with several WSN standards such as IEEE 802.11 [3], IEEE 82.15 [9] and IEEE 802.15.4 [10]. The standard IEEE 802.14 is designed to support low power consumption, low data rate and low cost for wireless communication. Also, there are various architectures based on this standard e.g. Zigbee, 6LoWPAN, WirelessHART, ISA100.11a. Ultra Wideband [11] and Z-wave [12] use a Radio Frequency Technology (RF) for wireless communication.

Considering for indoor solutions, some wireless communication technologies are good candidates such as ANT [13], Dash7 [14], EnOcean technology [15]. However, the mentioned technologies are difficult to be the key technology of WSN for IoT devices that need a variety of connection. Recently, BLE technology has been popular for IoTs because it enables a low cost, low power module and is typically implemented in commercial smartphones and sensors. Because of this reason, in our research, we employ BLE as the wireless communication module for a mesh sensor network.

To support the future work, which will apply ML in the mesh network, flooding based data propagation over the mesh network is needed. For flooding based data propagation over the mesh network, time scheduling management on sensor nodes is important. In the previous research [16], Mathias et al. proposed BLE mesh network using Nordic nRF52832 modules and evaluated the latency performance with different methodologies based on flooding. The result shows that the backoff mechanism has a lot of impact on the Round Trip Time (RTT) but makes the network less scalable and robust.

3. Mesh Network Architecture

3.1 Sensor node development

To develop mesh sensor network, we use PSoC4 BLE and 2JCIE-BL Omron environment sensor as a node in a network as shown in Fig.2. Omron environment sensor can sense acceleration, temperature, humidity, light, UV index, barometric pressure, noise of environment area within 10m range and send data to a receiver by BLE. The sensor operating voltage is from 2.7 to 3.3V which is very low power and needs only 3VDC Lithium battery to supply. Therefore, PSoC4 can receive data from Omron sensor periodically.

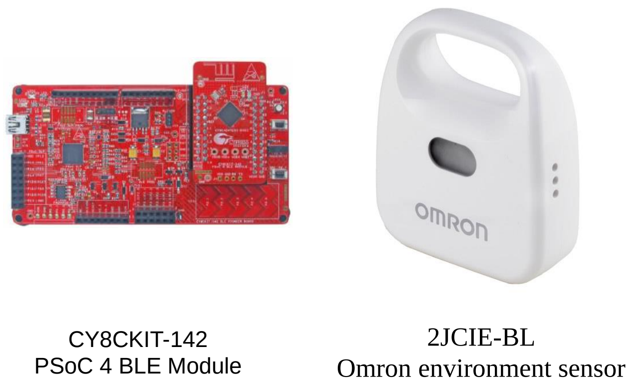

Fig. 2 Data processing architecture

3.2 Mesh network design and development

Our research utilizes the BLE mesh project provided on the Cypress webite. This project is a part of projects namely 100 projects in 100 days of Cypress Semiconductor. According to the example project, a payload is an RGB data. PSoC4 BLE will flood this data to other nodes when a user pushes the button on it. This example project is simply designed to transfer only one data

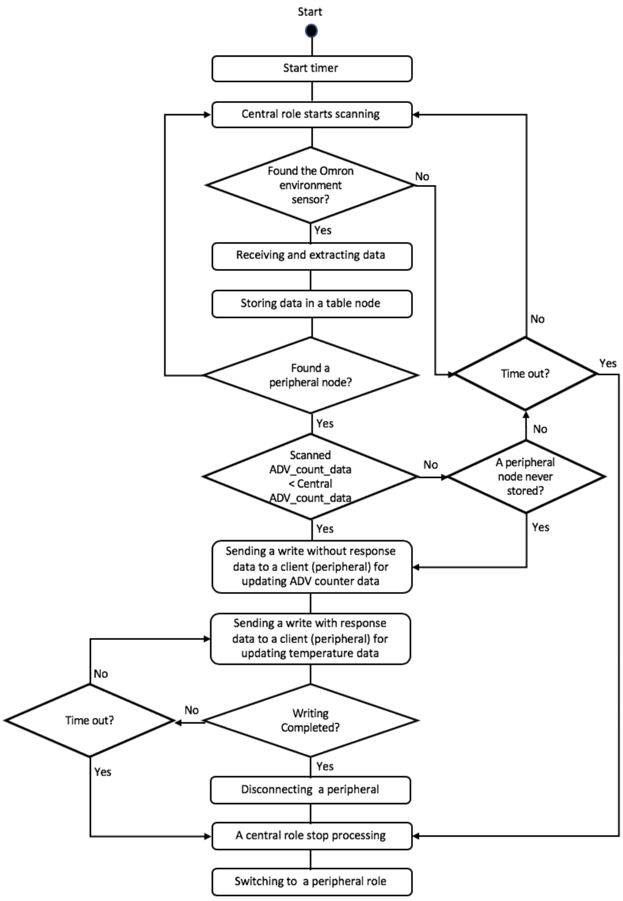

Fig. 3 A diagram of switching role in BLE mesh network

per period through the mesh network and the data is generated as a manual input from a user. Therefore, the modification in mesh network architecture is necessary to encourage each node periodically obtaining data from Omron environment sensor with BLE. Fig. 3 Illustrates a state diagram of mesh sensor architecture.

In order to apply the example to this research, PSoC4 BLE is initiated as a central role to scan and acquire data from Omron sensor. If it can receive data from sensor, the data will be extracted and only temperature data is updated into a node table. The duty of a node table is to implement every node in a network into individual rows to handle flooding-based transmission. Subsequently, a central node will continue scanning for a peripheral node and perform as a GATT server. In addition, each peripheral node advertises a counter data to be a GATT client. The counter data accumulates the number of receiving sensor data, then it will be implemented into advertised data.

The decision to transmit data depends on 2 cases of a scanned peripheral node: a peripheral node is used to store in a node table of central or a peripheral node never addresses in a node table. If the scanned peripheral node data is stored in the node table before, the central node known as a server will compare an advertised counter data of the scanned client in the peripheral node.

The central node will update the new advertised counter data as a writing data without a response packet in a case of an advertised data of the peripheral node is less than the one belonged to a central node. After that, the temperature data will be sent as a writing data with the response packet to update the temperature data in a node table. For the other case, the central can send data to a peripheral node to store the first data in a node table. After sending the payload, the node will change from a central role to a peripheral role for receiving data from other nodes.

Clearly, switching role between a central role and a peripheral role needs a precise scheduling management to decrease the packet loss rate in mesh sensor network. The next section describes a method to handle time scheduling in mesh network.

4. Time scheduling on Mesh Network

Fig. 4 shows a process of time scheduling on the mesh network. When all nodes are assigned to update their sensor data periodically, the scanning time of a central role node is initiated. The first task as a central role is to scan for Omron environment sensor node. If a node needs to complete updating data on mesh network without loss of any packet, the duration of a central role must be equal or more than the sensor updating and processing time. Since the first priority is that the central role node can obtain sensor data, the updating data process must be done before receiving the new data from Omron environment sensor. Considering this, the time duration of receiving sensor data (T_{sensor}) and the timeout duration of central role ($T_{central}$) are given by

$$T_{detect} + T_{ADV} + T_{update} \leq T_{central} < T_{sensor} \quad (1)$$

Because the data is an environment sensor data, the frequency of gaining sensor data can be low. T_{detect} is the time duration after starting scanning until detecting Omron environment sensor which can be affected by the amount of BLE modules in the area network. T_{ADV} is the time duration to discover a peripheral node until sending the updated counter data and T_{update} is the time duration from sending an updated temperature data until receiving a response. Considering $T_{central}$ of two neighboring nodes in mesh network, given τ is a timeout constant (i.e., duration of a central role). Then, the absolute timeout time of those central role nodes can be expressed as τ and 2τ, respectively (see Fig. 4).

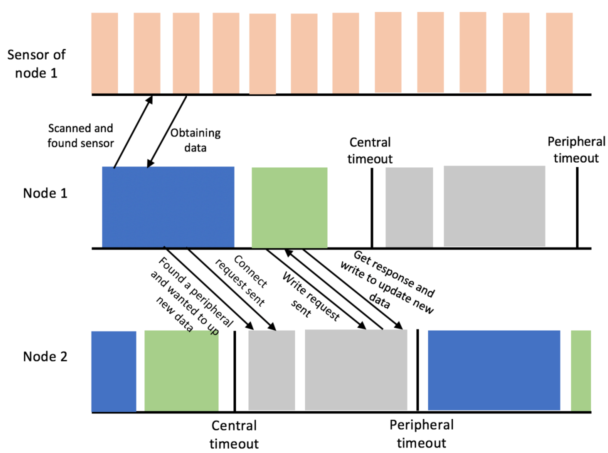

Fig. 4 Time scheduling of a node in mesh network

5. Test-bed Experiments

5.1 Experiment in a simple mesh network

As mentioned in the previous section, the experiment is set up by using PSoC4 BLE and Omron environment sensor to be a node of mesh network. To evaluate the proposed BLE mesh network, we start to construct a simple mesh network from 2 nodes. Therefore, each node just updates only its data through mesh network. In this case, the advertised counter data needs only 1 byte to transmit on the network. The temperature data contains 2 bytes of temperature data and 2 bytes are reserved for the future purpose, so the total of a payload is 4 bytes. In addition, we measure the packet loss rate by initiating $T_{central}$ at the different time from 50 ms to 3000ms while a watchdog timer of PSoC4 BLE is initiated at 100ms. In addition the experiment is located at a room which contains other BLE modules. We aggregated time when a central node can receive the Omron environment sensor data, time when a central found a peripheral node in mesh sensor network and time when a central node can connect and update a new data to a peripheral node. Also, we collected the connection time between a central and peripheral node with a number of an advertised data. In the experiment, the maximum number of data for transferring is 30 data. In the case of arriving packet is not equal and if any node reaches the maximum data, the experiment will be terminated, so we can measure a packet loss rate from this situation.

5.2 Results

This section explains results from the experiment. Since we measured individual timeout of a central role, the result of time to detect Omron environment sensor is shown in Fig. 5. As we can see that the average time to receive Omron environment sensor is not diverse except the timeout at 50ms. Examining a packet loss rate, only the time at 100ms and 50ms have a packet loss rate 20% and it performed only one role which a central node at 50ms is a peripheral role due to the receiving time is collected and a central node at 100ms also acks as a central role and only capable to send data through a network. To investigate the time when a central node connects to a peripheral node, we clearly see that at time 500ms and 1000ms finding a peripheral node time is lower than the others as demonstrated in Fig. 6. The connection time between a central node and a peripheral node is shown in Fig. 7. According to the figure, it indicates that the operation time of a peripheral role is significantly lower compared to a operation time of a central role. Especially, at time 50ms, 100ms, 500ms and 1000ms are dramatically lower than 1500ms. Furthermore, we measured the current consumption during a node changing its role as shown in Table 1. The current consumption when a node being a central node is greater than a peripheral role in every central operation time.

Current consumption (mA)		
Central	Peripheral	Central role operation time (ms)
16.38	6.52	50
17.30	8.64	100
20.34	9.77	500
18.70	7.73	1000
18.48	7.26	1500
20.38	9.87	3000
18.70	7.73	6000

Table 1 Table of Current consumption at different central role operation time

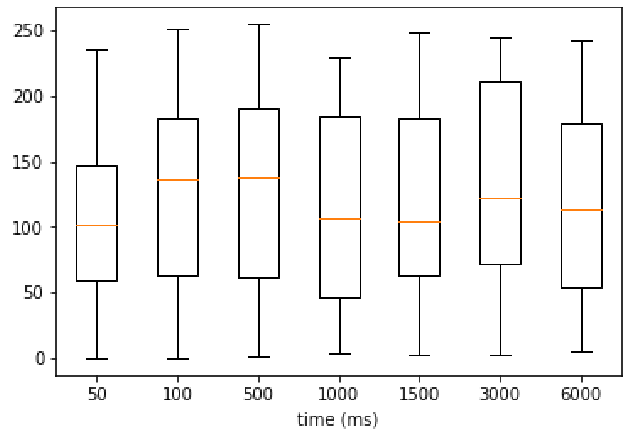

Fig. 5 Time of detecting Omron environment sensor at the different central timeout

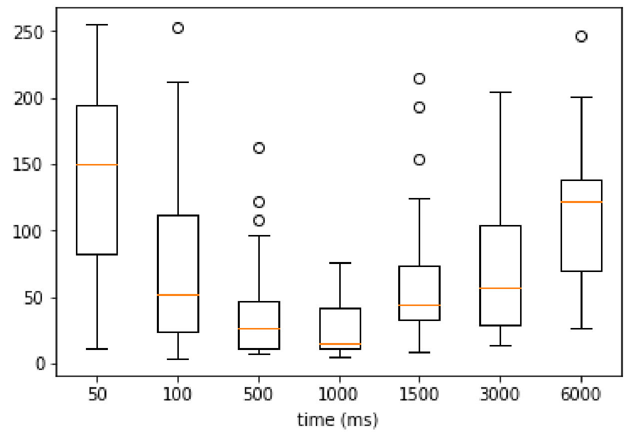

Fig. 6 Time when central found a node in mesh network at the different central timeout

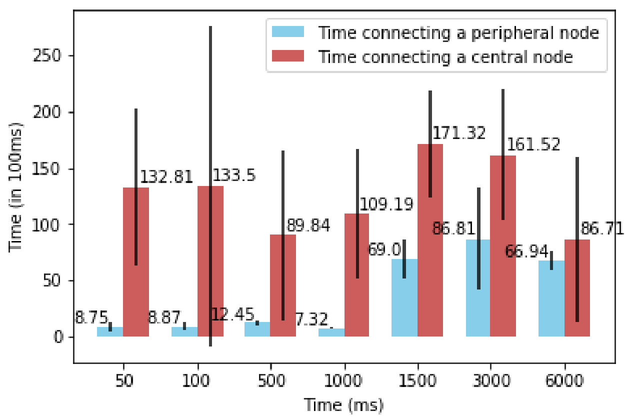

Fig. 7 Time connect between a central node and a peripheral node at the different timeout

5.3 Discussion

According to the result of an experiment, the time when a central discovering a node in network of 500ms and 1000 ms are greatly low compared to the time discovery node measured at other times. The reason of the unexpected result is that we assume that in the experimental room, normally there are more than 60 BLE modules. Therefore it is possible that the modules affect to the node discovery time. For this case, they may affect to BLE stack of a central node by storing PSoC4 BLE at the top of BLE stack of a central node. So, the central node at 500ms and 1000ms could be discovered by a central node rapidly. Moreover, a very low connection time of a peripheral node at 50ms, 100ms and 500ms result from the time initiation to a central. The lower time operation would be good for power consumption and fast processing. However, its performance is harmful to a mesh network because it may occur sending without response or incomplete procedure would lead to occur a network congestion. To solve this problem, we need to implement such a time scheduler as a background task but this will be the trade-off between a lot of factors such as computational time, memory resource and power consumption. Considering a result of current consumption among a central role and a peripheral role, the current consumption of a peripheral node is lower than 2 times of a central role due to a longer operation time of a central role. In addition, we noticed the current consumption slightly increases when a central node processed BLE scanning. If we examine from Table 1, the lowest current consumption is at 50ms of a central operation time. It means at this operation time meets the requirement of decreasing power consumption. However, because of the result of a packet loss rate, it is too difficult to select it as a proper time for operation.

5.4 Future work

In the future work, we would like to implement machine learning to learn such services to support people in daily life. A lot of additional procedures have to improve such as congestion controlling for mesh network, and energy harvesting module in order to support the self-generated power of nodes in a network. This energy harvesting module can be an important attribution of the next generation of WSN. Because this module deceasing the cost of maintenance and save the deployment cost. In the collaboration of our research has already started to develop an energy harvesting module Fig. 8 illustrates a development module that will be implement in the mesh sensor network.

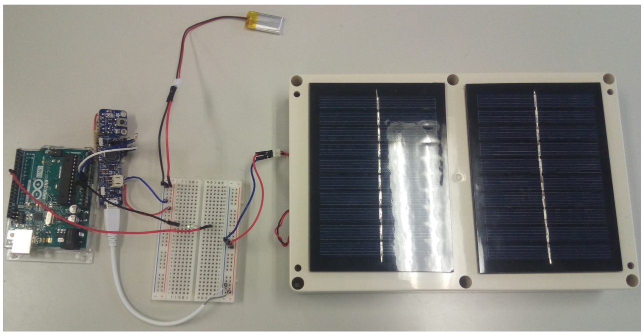

Fig. 8 The energy harvesting module

6. Conclusion

In this paper, we studies the feasibility to implement a MCU including BLE to receive an environment sensor as a node in mesh network. We proposed a renew paradigm to support a mesh topology by switching a role of BLE between a central role and a peripheral role. After that, we set up a test-bed experiment with a simple mesh network and measure the processing time at different operation time of a central role. The result of our research showing that it is possible to implement a BLE based mesh sensor network. Most of time measurement performed a possible method to support the real-world services and the measurement of current consumption shows the BLE-based mesh sensor network consuming a very low energy and suitable to continue study on it. However, more precisely time scheduling and a sophisticated wireless network handling should be further develop and study. Also, the additional part such an energy harvesting module to decrease power consumption should be the first priority to implement in as a node of mesh sensor network in the near future.

Acknowledgments

This work is supported by "Research and Development of Innovative Network Technologies to Create the Future," the Commissioned Research of National Institute of Information and Communications Technology (NICT), Japan.

References

[1] Taherkordi A., Mohammadi R., Eliassen F.: A communication-efficient distributed clustering algorithm for sensor networks, *Proceedings of the 22nd International Conference on Advanced Information Networking and Applications-Workshops (AINAW 2008)*, pp. 634–638 (2008).

[2] Li C., Xie X., Huang Y., Wang H., Niu C.: Distributed data mining based on deep neural network for wireless sensor network, *International Journal of Distributed Sensor Networks*, Vol. 11, No. 7, pp. 157453 (2015).

[3] Fukushima, Y., Miura, D., Hamatani, T., Yamaguchi, H. and Higashino T.:MicroDeep: In-network Deep Learning by Micro-Sensor Coordination for Pervasive Computing,*2018 IEEE International Conference on Smart Computing (SMARTCOMP)*, pp. 163-170, 2018.

[4] Morita, T., Fujiwara, M., Arakawa, Y., Suwa, H. andYasumoto, K.: Energy Harvesting Sensor Node TowardZero Energy In-Network Sensor Data Processing,*MobileComputing, Applications, and Services, Cham, SpringerInternational Publishing*, pp. 210–215 (2018).

[5] Bradley, J., Barbier, J., and Handler, D.: Embracing the Internet of Everything To Capture Your Share of $14.4 Trillion (White Paper). available from <`https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoE_Economy.pdf`>`(accessed2017-11-22)`.

[6] Yasumoto, K., Yamaguchi, H., and Shigeno, H.: Survey of Real-time Processing Technologies of IoT Data Streams. In: *Journal of Information Processing*, Vol.24, No.2, pp.195-202, 2016.

[7] Sarkar, C., Nambi, A.U., Prasad, R.V., Rahim, A., Neisse, R., and Baldini, G.: DIAT: A scalable distributed architecture for IoT. In: *IEEE Internet of Things Journal*, vo.2, no.3, pp.230-239, 2015.

[8] Nakamura, Y., Mizumoto, T., Suwa, H., Arakawa, Y., Yamaguchi, H., Yasumoto, K.: In-Situ Resource Provisioning with Adaptive Scale-out for Regional IoT Services, *The Third ACM/IEEE Symposium on Edge Computing (SEC 2018)*, Bellevue, USA, October 25-27, 2018.

[9] IEEE Standard 802.11 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, *IEEE STD 802.11*, (2007).

[10] IEEE Standard 802.11 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, *IEEE STD 802.11*, (2007).

[11] IEEE: IEEE 802.15 Working Group for WPAN (online), available from ⟨http://www.ieee802.org/15/⟩.

[12] IEEE Standard 802.15.4 Standard for Information Technology Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer

(PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs) (2006).
[13] Porcino D., Hirt W.: Ultra-wideband radio technology: potential and challenges ahead, *Communications Magazine, IEEE*, Vol. 41, No. 7, pp. 66–74 (2003).
[14] zen-sys: Z-Wave Alliance (online), available from ⟨https://www.z-wave.com/learn⟩.
[15] Dynastream Innovations Inc.: ANT Message Protocol and Usage (2014).
[16] Dash7 (online), available from ⟨http://www.dash7-alliance.org⟩.
[17] EnOcean (online), available from ⟨https://www.enocean.com/en/technology/energy-harvesting-wireless⟩.
[18] Mathias B., Jen R., Adnan S., Jeroen H.: The Bluetooth Mesh Standard: An Overview and Experimental Evaluation, *Sensors*, Vol. 18, No. 8, pp. 1424–8220 (2018).