Crowdedness Estimation Using RSSI on Already-deployed Wireless Sensor Networks

Naoya Matsumoto†, Jiei Kawasaki†, Makoto Suzuki‡, Shunsuke Saruwatari†, Takashi Watanabe†
†Graduate School of Information Science and Technology, Osaka University, Japan
‡Sonas Corporation, Japan

***Abstract*—This paper proposes a system that estimates crowdedness, such as the numbers of people and wireless communication devices, from the received signal strength indication (RSSI), which is measured using sensor nodes already-deployed for a different purpose. For example, the system estimates crowdedness using sensor nodes of structural health monitoring. Our system consists of two parts: an RSSI synchronous measurement and a crowdedness estimation algorithm. The RSSI synchronous measurement enables us to measure and collect two kinds of RSSI by synchronized RSSI sampling: the inter-node RSSI and surrounding RSSI. The crowdedness estimation algorithm estimates the number of people from the collected inter-node RSSI and the number of wireless communication devices from the collected surrounding RSSI in a location. We evaluated the proposed system in the laboratory at Osaka University. The evaluation results demonstrate that the system estimates the presence or absence of people in the laboratory with approximately 92% accuracy and the number of people with approximately 79% accuracy, with errors of up to two people.**

***Index Terms*—Crowdedness Estimation, RSSI, Synchronized Sampling, Sensor Networks, Wireless Sensing**

I. Introduction

With the advent of smartphones and the Internet of Things (IoT) devices, traffic using radio waves continues to increase significantly. The authors are working on a wireless sensing approach that senses a location and user information from radio waves that are transmitted by wireless communication devices. This wireless sensing approach enables us to integrate a communication system and sensing system.

This paper considers a crowdedness estimation using radio waves. We define the crowdedness estimation as estimating the number of people and wireless communication devices in a location. The estimated crowdedness facilitates situation-dependent services, such as energy consumption reduction in a smart building, navigation in a large shopping mall, observation of elderly people, and intrusion detection in facilities. The estimated crowdedness can also be utilized to improve wireless communication performance according to the number of devices in a wireless communication protocol.

Considering usage in daily life, the crowdedness estimation system needs to satisfy two requirements: a device-free approach and low deployment cost. Various crowdedness estimation methods using radio waves or cameras have been studied [1]–[12]. However, there is currently no approach satisfying the two abovementioned requirements. The details of these requirements and related work will be discussed in Section II.

In view of this, this paper proposes a crowdedness estimation using a synchronized RSSI measured by IEEE 802.15.4. First, we propose a mechanism to measure two kinds of RSSI: the inter-node RSSI and surrounding RSSI. Both of these RSSIs are strictly synchronized using "Choco" which is a wireless sensor network platform utilizing simultaneous transmission [13]–[15]. The proposed method enables us to estimate the crowdedness just by adding RSSI acquirement features to IEEE 802.15.4 wireless sensor networks already-deployed for a different purpose. Next, we propose crowdedness estimation algorithms that estimate the number of people from the inter-node RSSI and the number of devices from the surrounding RSSI. An experiment conducted in the laboratory confirms that the algorithm estimates the number of people with approximately 79% accuracy, with errors up to two people.

The remainder of this paper is organized as follows. Section II describes the requirements for crowdedness estimation and existing methods. Section III presents an overview of the proposed system. Section IV presents the mechanism for collecting data. We describe how to process the collected data in Section V. We evaluate the data collection rate and the proposed system in Section VI. Finally, our conclusions are given in Section VII.

II. Crowdedness estimation

A. Requirements for crowdedness estimation

In this paper, crowdedness estimation is defined as a technique to estimate how many users, or how many wireless communication devices exist in a location. The crowdedness estimation will enable us to utilize various applications in daily life. For example, according to the crowdedness, a smart building can reduce energy consumption by adjusting the air conditioning [16], a large shopping mall can reduce waiting times and optimize the arrangement of shops, and a communication protocol can improve the communication performance by adjusting the initial backoff time. Crowdedness estimation can also be applied to observation services for elderly people living alone and for intrusion detection in a house.

For such applications, a crowdedness estimation system must satisfy the following two requirements:

1) device-free estimation,
2) a low deployment cost.

Device-free estimation is desirable to reduce the user's burden. If a user requires a specific device, then problems such

as battery exhaustion and device failure may arise. A low development cost includes the financial and mental costs when deploying a crowdedness estimation system. It is desirable to deploy the system as cheaply and easily as possible to a target location. A low invasion of privacy is also desirable for users.

B. Existing crowdedness estimations

Two kinds of crowdedness estimation exist: estimation using the position information of all users and estimation from direct spatial information. Position estimation methods include GPS, Wi-Fi, and Bluetooth [1]–[5]. For example, [3] estimates user positions using the flight times of transmitted radio waves. However, estimations using GPS, Wi-Fi, or Bluetooth do not represent device-free approaches.

A crowdedness estimation from direct spatial information employs cameras or infrared ray sensors, and these are categorized as device-free approaches [6]–[8]. For example, [7] estimates the crowdedness by counting the number of people based on the visual features from images captured by cameras.

However, such camera-based approaches have some drawbacks, including the limitations of the deployment location, privacy, price, and computational cost of image processing. For example, a limitation of the deployment location results from the strong directivity of light. The directivity of light affects the acquisition range of a camera, as well as shadowing caused by obstacles.

Estimation using radio waves has also been studied as device-free approach [9]–[12]. For example, [11] estimates the crowdedness from radio waves acquired between two terminals deployed in a location. However, in that approach, it is necessary to conduct prior calibrations in the location, where all people walk at the same speed.

III. Overview of the Proposed System

To resolve the problems mentioned in Section II, we integrate a crowdedness estimation mechanism with already-deployed wireless sensor networks. The proposed mechanism senses the state of a location using only the RSSI acquired by already-deployed IEEE 802.15.4 sensor nodes [17]. Our approach is device-free and easy to deploy, because already-deployed IEEE 802.15.4 sensor nodes are available, such as for structural health monitoring, environmental monitoring, and smart meters [18]–[20].

Fig. 1 presents an overview of the proposed crowdedness estimation system. Sensor nodes estimate the crowdedness by evaluating the variance of the synchronized inter-node RSSI. The synchronized sampling maintains the time consistency of the RSSI acquired by different sensor nodes. The time consistency of the RSSI enables us to capture the radio wave environments of a location at a certain moment. The details of the synchronized sampling are presented in Section IV.

All sensor nodes send the RSSI to a sink node, and the sink node processes the RSSI to estimate the crowdedness. Users do not require a specific device for the crowdedness estimation. Our RSSI-based system has lower privacy issues compared to camera-based systems, because the sizes of terminals are small, and our system does not capture figures. The details of the crowdedness estimation using the RSSI are presented in Section V.

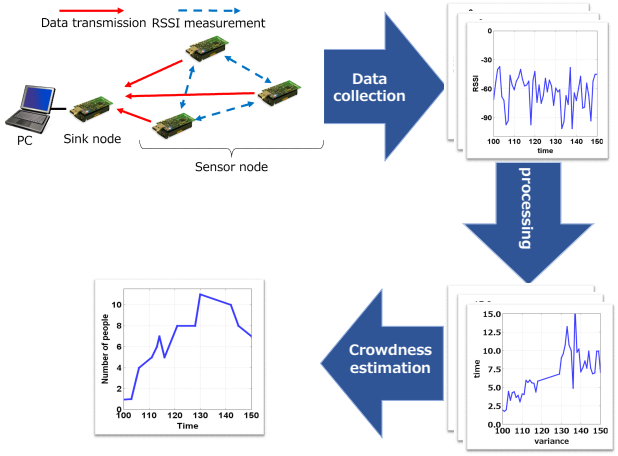

Fig. 1. Overview of the proposed system

IV. Synchronized Sampling System

The synchronized sampling system enables each sensor node to acquire two kinds of RSSI: the inter-node RSSI and surrounding RSSI. The inter-node RSSI is the strength of radio waves when a sensor node receives signals that another sensor node transmits. If a person is present between sensor nodes, then the inter-node RSSI decreases because of shadowing. This decrease enables us to capture the change in a location. Each node measures the strengths of non-modulated sine waves transmitted from another sensor node as the inter-node RSSI. A person's average walking speed is approximately 1.25 m/s, and a person's average width is approximately 30 cm. Sampling within 240 ms enables the system to capture the influence on the RSSI of the passage of people.

The surrounding RSSI is the strengths of radio waves when a sensor node receives signals that other sensor nodes do not transmit. When no sensor nodes are communicating, each sensor node measures the surrounding RSSI. The surrounding RSSI enables us to capture the communication states of wireless communication devices such as wireless LAN and Bluetooth.

A. Concurrent transmission flooding

The proposed system enables sensor nodes to measure the synchronized RSSI using "Choco" [13], [14]. The synchronized RSSI enables us to capture the radio wave environment of a location at a certain moment. Choco provides accurate time synchronization and reliable low-power data collection using concurrent transmission flooding [15].

Fig. 2 presents an example of concurrent transmission flooding. Concurrent transmission flooding transmits a packet to a destination node through simultaneous broadcasting by all nodes, rather than through single path transmission. The quadrangle indicates the sink node, the triangle represents a

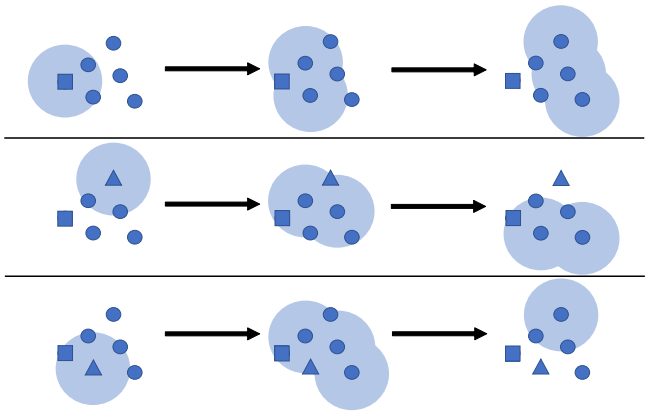

Fig. 2. Concurrent transmission flooding

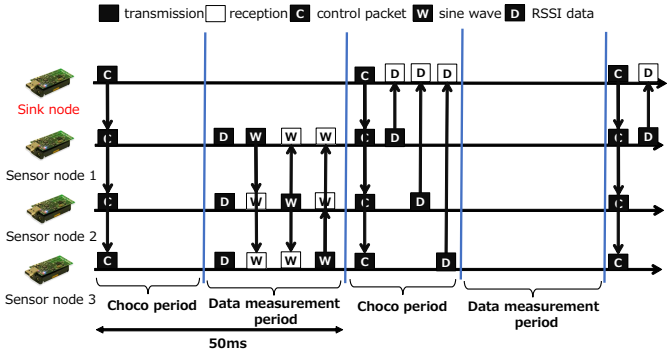

Fig. 3. Example of synchronized sampling operation

source node, and the circles denote relay nodes. The sink node transmits control packets to the other nodes. The control packets enable all nodes to schedule packet transmission timing. Upon receiving a packet, relay nodes immediately rebroadcast the packet. This simultaneous broadcasting enables a packet to spread in the entire network. Even when some nodes do not receive a packet, this chained re-broadcasting enables the packet to reach the destination node.

Choco achieves low power consumption by scheduling packet transmission according to time slots. A sink node utilizes a control packet to dictate the following four operations to sensor nodes:

1) "which slot a node transmits packets in,"
2) "which sequence number a node transmits packets in,"
3) "which node a node transmits packets to,"
4) "which data a node transmits."

Each sensor node transmits packets according to the schedule. Each packet from each sensor node has a backlog field in a header. A backlog represents the number of packets to transmit. The sink node calculates the number of packets to collect from a backlog field, and allocates slots to sensor nodes. If backlogs on all nodes are zero, then the sink node broadcasts a sleep packet. When receiving a sleep packet, a sensor node transits to sleep mode. Determining "which slot a node transmits packets in" and "which sequence number a node transmits packets in" enables sensor nodes to reduce the standby time.

B. Synchronized sampling

Concurrent transmission flooding is also utilized for synchronized sampling. A sink node transmits a synchronization packet to sensor nodes every second using concurrent transmission flooding. The synchronization packets enable sensor nodes to synchronize with 1 μs accuracy.

The proposed system uses synchronized sampling to acquire the RSSI. This synchronized sampling involves synchronizing all sensor nodes with the sink node and aligning the sampling timing among all sensor nodes. The synchronized sampling maintains the time consistency of the RSSI acquired by each sensor node.

Fig. 3 illustrates the operation synchronized sampling. Sensor nodes alternately measure the inter-node RSSI and surrounding RSSI through synchronized sampling every 200 ms. One slot consists of 50 ms, and each cycle is 200 ms. Each slot consists of the two periods: the Choco period and data measurement period. The Choco period is 15 ms, and the data measurement period is 35 ms. The sink node transmits a control packet and collects data in the Choco period. Sensor nodes measure the inter-node RSSI and surrounding RSSI in the data measurement period.

The sink node transmits the control packet to all sensor nodes at the beginning of each slot. The control packet includes the schedule of sensor nodes. The sink node strictly manages the operation of sensor nodes using the time synchronization and reliable transmission. This management enables sensor nodes to improve power efficiency.

Sensor nodes measure the inter-node RSSI and surrounding RSSI in the data measurement period of the first slot on each cycle. First, no sensor nodes transmit any packets, and they measure the surrounding RSSI. Each sensor node stores the surrounding RSSI. Next, sensor nodes take turns broadcasting unmodulated sine waves and measuring the inter-node RSSI. Each sensor node stores the source node's ID and the inter-node RSSI. The source node's ID is determined from the transmission time, according to the schedule in the control packet. Sensor nodes transmit the RSSI to the sink node using concurrent transmission flooding.

Sensor nodes transmit the RSSI to the sink node in the remaining three slots of the Choco period. A sensor node that receives the control packet transmits the inter-node RSSI and surrounding RSSI to the sink node, according to the schedule in the control packet. Sensor nodes do not measure the RSSI and transit to sleep mode in the remaining three slots of the data measurement period.

V. Crowdedness Estimation using the RSSI

We propose the two kinds of crowdedness estimation: estimations of the number of people and the number of devices. The inter-node RSSI enables us to estimate the number of people, and the surrounding RSSI enables us to estimate

the number of devices. These RSSIs are measured by the synchronized sampling method detailed in Section IV-B.

A. Estimation of the number of people

Algorithm 1 Estimation of the number of people

1: **for** x in $\mathbb{S}$ **do**
2: **for** y in $\mathbb{S}$ **do**
3: $\sigma^2_{\rm sum} \Leftarrow \sigma^2_{\rm sum} + V(x, y, T)$
4: **end for**
5: **end for**
6: $\overline{\sigma^2} \Leftarrow \frac{\sigma^2_{\rm sum}}{|\mathbb{S}|^2}$
7: **if** $\overline{\sigma^2} \leq \gamma$ **then**
8: $u = 0$
9: **else**
10: $u \Leftarrow \lfloor \alpha \times \overline{\sigma^2} + \beta + 0.5 \rfloor$
11: **end if**

TABLE I
VARIABLES USED IN **Algorithm 1**

variable name	role in the algorithm
$\mathbb{S}$	the set of sensor nodes
$V(x, y, T)$	the variance of inter-node RSSI between x and y
$\overline{\sigma^2}$	the average of the variance of inter-node RSSI

We estimate the number of people using the variance of the inter-node RSSI. The variance of the inter-node RSSI enables us to reduce the environmental influence. The inter-node RSSI changes depending on the temperature, humidity, and movement of people. The fluctuation of the inter-node RSSI owing to user passage is large and instantaneous, and fluctuations owing to temperature and humidity are small and gentle. In addition, the synchronized sampling approach from Section IV-B enables us to distinguish whether a user moves or multiple users are present.

Algorithm 1 describes the estimation of the number of people. Table I lists the variable names and roles in **Algorithm 1**. Here, $\mathbb{S}$ is the set of sensor nodes, $\sigma^2_{\rm sum}$ is the variable for calculation, $\overline{\sigma^2}$ is the average of the variance of the inter-node RSSI between all nodes, u is the estimated crowdedness, and $V(x, y, T)$ is a function for calculating the variance of the inter-node RSSI between the sensor nodes x and y for T seconds.

Lines 1 to 6 in **Algorithm 1** calculate the variance of the inter-node RSSI. The variance of the RSSI on each path is calculated individually, and all of the variances are averaged over. Lines 7 to 9 in **Algorithm 1** estimate whether or not a person is present. If the variance of the inter-node RSSI is less than γ, then it is estimated that there is no person, and 0 is substituted into u. If the variance of the inter-node RSSI is greater than or equal to γ, then **Algorithm 1** substitutes the rounded value of $\alpha \times \overline{\sigma^2} + \beta$ into u. In addition, γ is a constant calculated from experimental data and α and β are constants calculated by the least-squares method, using the variance of the inter-node RSSI and the actual number of people.

B. Estimation of the number of devices

We estimate the number of devices using a clustering algorithm with the surrounding RSSI. If wireless devices communicate, then the surrounding RSSI close to the devices increases probabilistically. The minimum time required for transmitting and receiving one packet in Wi-Fi is approximately 20 μs. The sampling interval of each sensor node is 200 ms.

TABLE II
VARIABLES USED IN **Algorithm2**

variable name	role in algorithm
$\mathbb{S}$	the set of sensor nodes
$D(x, t)$	node x's surrounding RSSI
$\overline{D\mathrm{sum}(\mathbb{S}, t)}$	the average of surrounding RSSI
$\min(D(\mathbb{S}, t))$	the minimum surrounding RSSI
$\mathrm{xmeans}(N(\mathbb{S}, \mathbb{T}))$	the number of total clusters

The total number of surrounding RSSI patterns may represent the number of devices. For example, if there is one active device in a location, then the surrounding RSSI is divided into two patterns: the case that all sensor nodes measure a packet from a device and that no sensor nodes measure any packets from a device. If the number of devices in a location increases, then each sensor node measures many patterns of the surrounding RSSI.

To calculate the total number of surrounding RSSI patterns, the estimation employs the x-means method [21]. The x-means method is a clustering algorithm that automatically determines the total number of clusters using the Bayesian information criterion (BIC). The x-means method performs clustering by repeating the following two processes:

1) divide each cluster into two and calculate the BIC for each cluster,
2) perform 1) if the BIC after this division is smaller than that before the division, and stop 1) if not.

Algorithm 2 Estimation of the number of devices

1: **for** x in $\mathbb{S}$ **do**
2: $D\mathrm{sum}(t) \Leftarrow D\mathrm{sum}(t) + D(x, t)$
3: **end for**
4: $\overline{D\mathrm{sum}(t)} \Leftarrow \frac{D\mathrm{sum}(t)}{|\mathbb{S}|}$
5: **if** $\overline{D\mathrm{sum}(\mathbb{S}, t)} > \gamma$ **then**
6: $\epsilon \Leftarrow \min(D(\mathbb{S}, t))$
7: **for** x in $\mathbb{S}$ **do**
8: $N(x, t) \Leftarrow D(x, t) - \epsilon$
9: **end for**
10: **end if**
11: $\theta \Leftarrow \lambda \cdot \mathrm{xmeans}(N(\mathbb{S}, \mathbb{T}))$

Algorithm 2 presents the estimation of the number of devices every T seconds. Table II lists the variable names and roles in **Algorithm 2**. Here, $\mathbb{S}$ is the set of sensor nodes, $\mathbb{T}$ is the set of elapsed times, t is an element of $\mathbb{T}$, $D(\mathbb{S}, t)$ is the surrounding RSSI acquired by the sensor nodes at t, ϵ and $D\mathrm{sum}(t)$ are variables for calculation, $\overline{D\mathrm{sum}(\mathbb{S}, t)}$ is the average of the surrounding RSSI at t, $N(\mathbb{S}, t)$ is the normalized surrounding RSSI at t, θ is the estimated number of devices, $\min(D(\mathbb{S}, t))$ is a function that calculates the minimum value of the surrounding RSSI at t, and $\mathrm{xmeans}(N(\mathbb{S}, \mathbb{T}))$ is a function that calculates the total number of clusters.

Line 5 in **Algorithm 2** removes noise from the data. If the average of the surrounding RSSI is less than or equal to

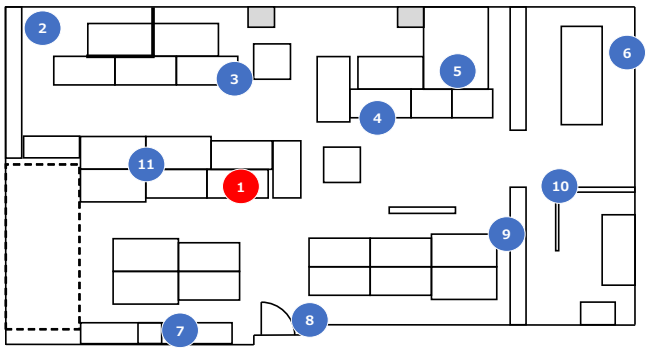

Fig. 4. Development positions of the sink node and sensor nodes

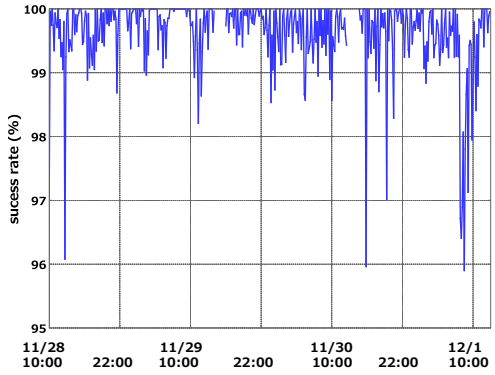

Fig. 5. Success rate of data collection

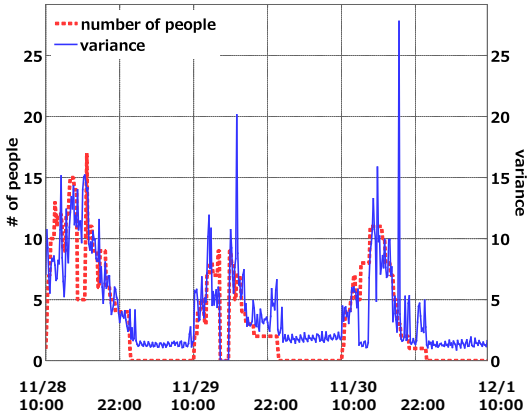

Fig. 6. Time series of the variance of the inter-node RSSI and number of people

γ, then **Algorithm 2** removes this data, because data below γ is regarded as noise. If there are no data transmitting devices, then the surrounding RSSI becomes small. Here, γ is a constant calculated from experimental data. Line 8 in **Algorithm 2** normalizes the surrounding RSSI. The normalization of the surrounding RSSI enables the system to make the base value at every time equal to zero and classify the data for each pattern. The normalization of the surrounding RSSI consists of subtracting the minimum surrounding RSSI from the surrounding RSSI of each sensor node.

Line 11 in **Algorithm 2** estimates the number of devices. The total number of clusters represents the total number of surrounding RSSI patterns. Here, λ is a constant calculated by comparing the total number of clusters by an experiment in an anechoic chamber with the actual number of active devices.

VI. Evaluation

A. Experimental settings

We conducted an experimental evaluation in the laboratory at Osaka University from 10:00 on November 28th, 2017 to 10:00 on December 1st, 2017. We installed a sink node and 10 sensor nodes in the laboratory. A Sonas CC-RM01 was used as the sink node [22], and Choco evaluation boards based on Texas Instruments CC2650MODA were utilized as sensor nodes. A Buffalo WZR-1750DHP2 was employed as an access point. The radio channel was 13 (2.472 GHz). Fig. 4 illustrates the development locations of nodes. The sink node was placed at 1, and the sensor nodes were placed at positions 2 to 11. A laboratory meeting was held from 15:00 to 16:30 on November 28th. Because the laboratory meeting was held in a separate room, the number of people in the laboratory rapidly decreased.

B. Success rate of data collection

Fig. 5 illustrates the data collection rate in the experiment. The average success rate of the data collection was 99.6%. The horizontal axis represents the time from 10:00 on November 28th, 2017 to 10:00 on December 1st, 2017, and the vertical axis represents the success rate of the data collection. The success rate continually exceeded 95%. The cause of packet drops was interference from other wireless networks. For example, the surrounding RSSI drastically increased owing to the download of meeting documents between 15:00 and 16:30 on November 28th.

C. Evaluation of estimating the number of people

Fig. 6 illustrates the variance of the inter-node RSSI and the actual number of people. The horizontal axis represents the time from 10:00 on November 28th, 2017 to 10:00 on December 1st, 2017. The vertical axis on the right side represents the variance of the inter-node RSSI, and the vertical axis on the left side shows the number of people in the laboratory. Fig. 6 confirms two observations. The first is that the variance is large when people are present, and small when they are not. We can observe that the variance is increasing when the number of people is increasing at around 10:00. The correlation coefficient between the number of people and the variance of the inter-node RSSI is approximately 0.71. The second observation is that the variance is too large compared to the actual number of people between 15:00 and 16:30 on November 28th. Radio waves from other wireless networks, such as Wi-Fi, might influence the inter-node RSSI.

Next, we evaluated the crowdedness estimation by comparing the variance of the inter-node RSSI with the actual number of people. Here, γ, α, and β in **Algorithm 1** were set to -90, 0.83, and 0.65, respectively, and T was 600. Fig.

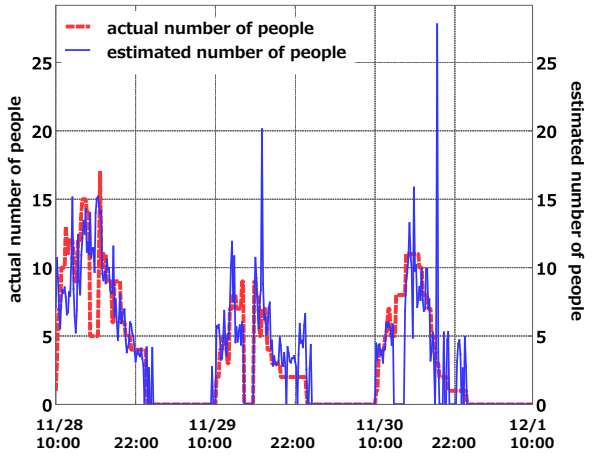

Fig. 7. Time series of the estimated and actual numbers of people

7 illustrates the estimated and actual numbers of people. The horizontal axis represents the time from 10:00 on November 28th, 2017 to 10:00 on December 1st, 2017. The vertical axis on the right side shows the estimated number of people, and the vertical axis on the left side represents the actual number of people. Fig. 7 confirms two observations. The first is that the algorithm estimated the presence or absence of a person with 92% accuracy. When people are present, an estimation error tends to occur. In the case with people, the algorithm correctly estimates their presence with 87% accuracy. When there are no people, the algorithm estimates their absence with 97% accuracy. The second observation is that the algorithm estimates the number of people with approximately 56% accuracy when δ is zero and with approximately 79% accuracy when δ is two. When the variance is large, an estimation error tends to occur.

D. Evaluation of estimating the number of devices

To evaluate the effectiveness of the algorithm presented in Section V-B, we conducted an experimental evaluation. We placed three PCs in an anechoic chamber. The three PCs were Panasonic CF-B11QWABR models with Ubuntu 16.04 LTS installed. One PC was wire-connected, and made into an access point. The radio channel was 1 (2.412 GHz). We used iperf, which is a network measurement tool, to transmit radio waves from one PC to another at approximately 10 Mbps. Because radio waves from the outside do not reach the anechoic chamber, the surrounding RSSI is only influenced by radio waves transmitted from the three PCs.

Fig. 8 presents the time series of the total number of clusters. The horizontal axis represents the elapsed time in the experiment, and the vertical axis shows the total number of clusters estimated by the x-means method. The estimation was performed every 10 min. The total number of clusters fluctuated between 10 and 15, although the number of active devices is constant. This confirms that the proposed algorithm does not perform effectively in estimating the number of devices.

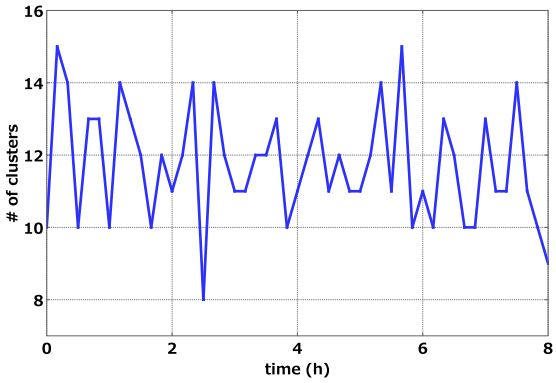

Fig. 8. Time series of the total number of clusters

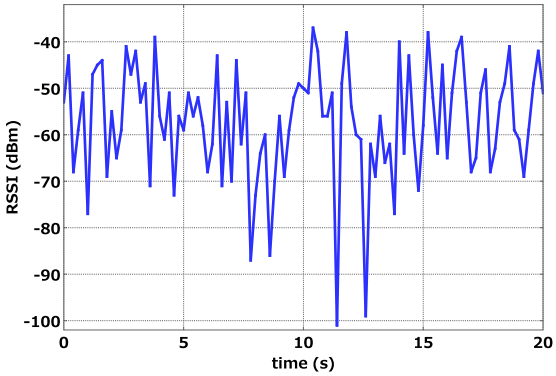

Fig. 9. Time series of surrounding RSSI acquired by one sensor node

Fig. 9 shows the time series of the surrounding RSSI acquired by one sensor node. The horizontal axis represents the elapsed time in the experiment, and the vertical axis shows the surrounding RSSI acquired by one sensor node. We assumed that a value of -90 dBm or less represents no transmitted radio waves. Fig. 9 confirms that there is a fluctuation of approximately ± 10 dBm around -60 dBm when a sensor node measures packets.

The fluctuation of the surrounding RSSI is caused by averaging the RSSI using a wireless module on the sensor node. IEEE 802.15.4 employs the RSSI for clear channel assessment (CCA). IEEE 802.15.4 often considers the average value over 128 μs rather than the instantaneous value of the RSSI. To determine whether a fluctuation of the surrounding RSSI occurred by calculating the average of the RSSI, we measured the surrounding RSSI using a power sensor that acquires the instantaneous RSSI. Linear Technology LT5534 was employed as a power sensor.

Fig. 10 illustrates the RSSI acquired by the power sensor. The horizontal axis represents the elapsed time in the experiment, and the vertical axis shows the surrounding RSSI acquired by the power sensor. Fig. 10 confirms that the surrounding RSSI is divided into three values: -65 dBm, -50 dBm, and -30 dBm. The power sensor does not measure an RSSI smaller than -65 dBm, because of the dynamic range

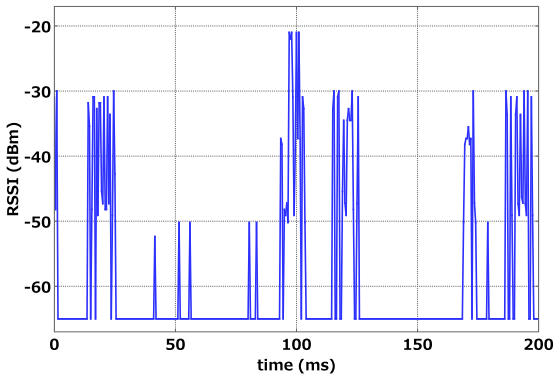

Fig. 10. Time series of the surrounding RSSI acquired by a power sensor

of the power sensor. Values of -50 dBm are considered to represent beacons and ACKs of the receiving PC, because the number of -50 dBm is small. Values of -30 dBm are considered to represent the radio waves of the transmitting PC, because the number of -30 dBm is large.

VII. Conclusion

This paper proposed a system to estimate the crowdedness, namely the numbers of people and wireless communication devices, in a location where a sensor network is deployed. Using "Choco", we proposed a mechanism to acquire and collect two kinds of synchronized RSSI: the inter-node RSSI and surrounding RSSI. We proposed and evaluated two algorithms, to estimate the number of people from the inter-node RSSI and the number of wireless communication devices from the surrounding RSSI. An experimental evaluation confirmed that our algorithm succeeded in estimating the number of people. On the other hand, we observed some problems, such as averaging the RSSI in a wireless chip, in the estimation of the number of devices.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Number JP16H01718 and JP17KT0042.

References

[1] U. Blanke, G. Troster, T. Franke, and P. Lukowicz, "Capturing crowd dynamics at large scale events using participatory GPS-localization," in *Proceedings of the 2014 International Conference on Intelligent Sensors, Sensor Networks and Information Processing (IEEE ISSNIP'14)*, Singapore, Apr. 2014, pp. 1–7.

[2] J. Biswas and M. Veloso, "WiFi localization and navigation for autonomous indoor mobile robots," in *Proceedings of the 2010 International Conference on Robotics and Automation (IEEE ICRA'10)*, Anchorage, AK, USA, May 2010, pp. 4379–4384.

[3] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single WiFi access point." in *Proceedings of the 2016 Networked Systems Design and Implementation (USENIX NSDI'16)*, Santa Clara, CA, USA, Mar. 2016, pp. 165–178.

[4] J. Weppner and P. Lukowicz, "Bluetooth based collaborative crowd density estimation with mobile phones," in *Proceedings of the 2013 Pervasive computing and communications (IEEE PerCom'13)*, San Diego, CA, USA, Mar. 2013, pp. 193–200.

[5] L. Chen, H. Kuusniemi, Y. Chen, J. Liu, L. Pei, L. Ruotsalainen, and R. Chen, "Constraint kalman filter for indoor bluetooth localization," in *Proceedings of the 23rd European Signal Processing Conference (EUSIPCO'15)*, France, Sep. 2015, pp. 1915–1919.

[6] Y. Yuan, C. Qiu, W. Xi, and J. Zhao, "Crowd density estimation using wireless sensor networks," in *Proceedings of the 2011 International Conference on Mobile Ad-hoc and Sensor Networks (IEEE MSN'11)*, Beijing, China, Dec. 2011, pp. 138–145.

[7] N. C. Tang, Y.-Y. Lin, M.-F. Weng, and H.-Y. M. Liao, "Cross-camera knowledge transfer for multiview people counting," *IEEE Transactions on image processing*, vol. 24, no. 1, pp. 80–93, Jan. 2015.

[8] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007.

[9] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763–776, 2017.

[10] K. Ohara, T. Maekawa, Y. Kishino, Y. Shirai, and F. Naya, "Transferring positioning model for device-free passive indoor localization," in *Proceedings of the 2015 International Joint Conference on Pervasive and Ubiquitous Computing (ACM UbiComp'15)*, Osaka, Japan, Sep. 2015, pp. 885–896.

[11] S. Depatla and Y. Mostofi, "Crowd counting through walls using WiFi," in *Proceedings of the 2018 International Conference on Pervasive Computing and Communications (IEEE PerCom'18)*, Athens, Greece, Aug. 2018, pp. 1–10.

[12] S. Depatla, A. Muralidharan, and Y. Mostofi, "Occupancy estimation using only WiFi power measurements," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 7, pp. 1381–1393, May 2015.

[13] M. Suzuki, Y. Yamashita, and H. Morikawa, "Low-power, end-to-end reliable collection using glossy for wireless sensor networks," in *Proceedings of the 2013 Vehicular Technology Conference (IEEE VTC'13)*, Dresden, Germany, Jun. 2013, pp. 1–5.

[14] M. Suzuki, C.-H. Liao, S. Ohara, K. Jinno, and H. Morikawa, "Wireless-transparent sensing," in *Proceedings of the 2017 International Conference on Embedded Wireless Systems and Networks (ACM EWSN'17)*, Uppsala, Sweden, Feb. 2017, pp. 66–77.

[15] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh, "Efficient network flooding and time synchronization with glossy," in *Proceedings of the 2005 International Conference on Information Processing in Sensor Networks (IPSN'05)*, Chicago, IL, USA, 2011, pp. 73–84.

[16] Y. Agarwal, B. Balaji, R. Gupta, J. Lyles, M. Wei, and T. Weng, "Occupancy-driven energy management for smart building automation," in *Proceedings of the 2nd ACM workshop on embedded sensing systems for energy-efficiency in building (ACM BuildSys'10)*, Zurich, Switzerland, Nov. 2010, pp. 1–6.

[17] IEEE 802 Working Group and others, "IEEE standard for local and metropolitan area networks-part 15.4: Low-rate wireless personal area networks (LR-WPANs)," *IEEE Std*, vol. 802, pp. 4–2011, Sep. 2011.

[18] M. Z. A. Bhuiyan, G. Wang, J. Wu, J. Cao, X. Liu, and T. Wang, "Dependable structural health monitoring using wireless sensor networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 4, pp. 363–376, Jul. 2017.

[19] G. Mois, S. Folea, and T. Sanislav, "Analysis of three IoT-based wireless sensors for environmental monitoring," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 8, pp. 2056–2064, Aug. 2017.

[20] M. J. Mudumbe and A. M. Abu-Mahfouz, "Smart water meter system for user-centric consumption measurement," in *Proceedings of the 2015 International Conference on Industrial Informatics (IEEE INDIN'15)*, Cambridge, UK, Jul. 2015, pp. 993–998.

[21] T. Ishioka *et al.*, "An expansion of x-means for automatically determining the optimal number of clusters," in *Proceedings of the 2005 International Conference on Computational Intelligence*, Alberta, Canada, Jul. 2005, pp. 91–95.

[22] Sonas. [Online]. Available: https://www.sonas.co.jp/en/