Report Timing Control Method for Monitoring WSN Applications
Jaesung Park and Changyong Yoon

Abstract
Wireless sensor network (WSN)-based monitoring application services frequently involve random installation of multiple sensor nodes in the monitoring area, allowing the nodes to transmit data periodically. However, this generates unnecessary and redundant data, causing an increase in data packet collision. This is because multiple sensors installed at similar locations will transmit the same data. Thus, this study proposes a method that enables each sensor node to autonomously determine data transmission time using only its local information. Through simulations, we verified that the proposed technique could control both unnecessary and redundant data transmission and data collision.

Keywords: Desynchronization, Autonomous control, WSN-based data collection

1. Introduction
A wireless sensor network (WSN) acquires information from the environment by using small sensor nodes, which have both low processing performance and storage capacity. The acquired information is then transmitted to the remote site through the sink node. Thus, WSNs have been utilized as a cost-effective method for monitoring areas that are either inaccessible by humans or are economically unmanageable [1-5]. These application services analyze and utilize the data that are periodically received from the monitoring areas through the sensor nodes. The data acquired through a WSN is transmitted to a data analysis server at constant intervals to ensure the accuracy of the decisions made by these application services [6].

The transmission power of a sensor node is limited, and the wireless communication channel between the sensor and sink nodes is unreliable. Because sensor nodes having data to send compete for radio resources using the carrier sense multiple access (CSMA), it is challenging to ensure reliable and accurate data transmission between the sensor node and the sink node [7]. Moreover, in contrast to mobile communication networks, where a service provider installs a base station along with controllers for a specific purpose, it is difficult to install sensor nodes at a specific location in a WSN used for monitoring surrounding areas. Therefore, the method of randomly installing multiple sensor nodes in the monitoring area has been frequently adopted as a solution, which allows the sensor nodes to transmit the data periodically. This method increases the probability of necessary data being acquired at appropriate times, because multiple sensors are installed at similar positions and the sensors are allowed to monitor the same area. However, this method generates unnecessary and redundant data because multiple sensors transmit the same data.
A clustering technique was proposed for collecting data reliably in a WSN [8]. The clustering method forms a cluster of sensor nodes to reduce the number of nodes competing for radio resources. However, message exchange between the sensor nodes is required for cluster configuration. When the signaling load is not properly managed, the resources of the sensor nodes are utilized more for control information transmission than for data transmission, which is problematic. A data reduction mechanism is adopted in a wide area network (WAN) to avoid packet collision and unnecessary retransmissions [9]. However, because each sensor node is an autonomous system, it is difficult to prevent all nodes from sending similar data simultaneously. The mobile sink technique, which involves the sink node moving and collecting information from the sensor node, can improve the reliability of data transmission between the sensor and sink nodes. However, it is challenging to optimize the trajectory of the sink node in an environment where the sensor nodes are randomly installed, or when a mobile sink cannot be used, depending on the type of monitoring application service. To improve the reliability of data reception and the accuracy of the reception time, the time division multiple access (TDMA) method was proposed as an alternative to the contention-based multiple access methods (such as IEEE 802.15.4 CSMA). However, the TDMA method requires time synchronization between the sensor nodes and shows a low adaptability to changes in the network topology [10]. Therefore, a desynchronization-based sensor data transmission technique, which uses a pulse-coupled oscillator (PCO) model, was proposed to solve the aforementioned problems with TDMA [9]. This technique eliminates data collisions by allowing each sensor to send one message at $T/n$ intervals in a distributed manner; the transmission period required by the application service is $T$, and the number of sensor nodes is $n$. However, because data transmission from the sensor to the sink is time-consuming, if $T$ decreases, or the number of sensor nodes increases, data collision occurs between the sensors. Moreover, because all sensor nodes transmit data once every period $T$, even if the transmission interval of each node is controlled, the generation of redundant data, which is unnecessarily increased, cannot be controlled, as the number of nodes increases.

Accordingly, we propose a method where each sensor node autonomously determines data transmission timing using only its local information. The proposed technique has the following three main design requirements. First, the generation of unnecessary and redundant data should be controlled by adjusting the amount of redundant data being generated within the WSN. Second, to satisfy the requirements of the application services, the WSN should reliably transmit at least one packet of data, for each data reception period. Third, message exchange is not required between any nodes for implementing the algorithm applicable to small sensor nodes. To satisfy these requirements, we develop an algorithm that can allow each sensor node to determine the data transmission timing autonomously, without exchanging control information with other nodes, based on the synchronization phenomena observed in nature [12][14].

This remaining part of this paper is as follows. In Section 2, we analyze the problem of data transmission in a WSN when used for monitoring application services. In Section 3, we describe a distributed algorithm for determining the data transmission timing of a sensor node, and further present the characteristics of the proposed algorithm. In Section 4, through simulation studies, we validate the proposed technique by comparing the performances of the proposed method with those of conventional methods with respect to the transmission time interval of the data transmitted from the WSN, the data transmission time interval of each sensor node, and the fairness among the sensor nodes in terms of the number of data transmissions. In Section 5, we conclude this study and propose the direction of future studies.

2. System Model and Problems

This study examines an environment where $n$ sensor nodes are installed to monitor the same information around a single sink. Each sensor node transmits the detected information to the sink every period ($T$) according to the requirements of the application service. Then, the sink transmits it to the server, which is located at the remote site. This study focuses on the packet transmission between the sensor and the sink, and assumes that the sink is connected to the Internet through reliable links, such as Ethernet or LTE, for transmitting information to the server.

Suppose multiple nodes transmit packets every period $T$ for monitoring the surrounding areas, the probability of the packet reception of the sink can increase, while numerous unnecessary redundant messages occur between the sensor and the sink (Figure 1(a)). This increases the probability of packet collisions between adjacent sensors, because the number of sensor nodes performing transmission at similar times increases. Furthermore, sensor nodes, which are not adjacent to each other, act as hidden nodes to each other. The probability of packet collisions increases as the number of hidden nodes grows (Figure 1(b)). Moreover, this unnecessary packet transmission drains...
the energy of the sensors, thereby shortening the operation time of the sensor. Therefore, the number of packets transferred from a WSN to the server every period \( T \) should be controlled to a reasonable level by considering the reliability of data transmission. In addition, as a sensor typically has a low data-processing capacity and small storage space, a simple algorithm that involves convenient implementation and does not need to exchange control information among any nodes is required.

The desynchronization model regards \( n \) nodes communicating with each other as an oscillator having the same frequency and period, but different phase angles. When they oscillate at the same frequency, the difference in phase angle between adjacent nodes becomes equal \([12, 14]\). Assuming that the phase angle of a node \( i \) is denoted as \( \phi_i \in [0, 1] \), the adjacent node is defined as a node with the largest phase angle among the nodes with an angle smaller than \( \phi_i \). The desynchronization model operates as follows. The node with the largest phase angle among \( n \) nodes (i.e., node \( i \)) is searched for, and the phase angle of all the nodes is increased by \( (1 - \phi_i) \). Thus, when \( \phi_i = 1 \), node \( i \) is initialized with \( \phi_i = 0 \), and \( i \) informs the other nodes that its phase angle is initialized to zero. Every node or some of the nodes that recognize that the phase angle of node \( i \) has been initialized, increase or decrease their phase angle. If this process is repeated, the phase angles of the nodes become desynchronized, and various desynchronization phenomena can occur depending on the degree to which the phase angle is increased or decreased, and in which node \([13]\).

3. Proposed Method

3.1 Algorithm

We denote the phase angle of node \( i \) as \( \phi_i(k) \) when a sensor node \( i \) receives the \( k \)-th message in a WSN. We also denote the residual time until a node \( i \) fires (i.e., send a message) as \( r_i(k) \).

Figure 2. Sensor node behavior.

**Initialization**
- Configure \( T, \phi_i(0) = \text{rand}(0, 1) \)
- Set \( r_i(0) = \phi_i(0) \cdot T, k = 0 \)

**When receiving a firing message from another node**

\[
\phi_i(k) = (1 - \alpha)\phi_i(k - 1), k + +
\]

\[
r_i(k) = r_i(k - 1) + \phi_i(k) \cdot T
\]

**When \( r_i(k) = 0 \)**

Node fires

- Set \( \phi_i(k) = 1, r_i(k) = T \)
initial waiting time as \( r_i(0) = \phi_i(0) \cdot T \). Whenever each node receives a fire message from other nodes, that node updates its phase angle with \( \phi_i(k) = (1 - \alpha) \phi_i(k - 1) \), and increases its waiting time with \( r_i(k) = r_i(k - 1) + \phi_i(k) \cdot T \). If \( r_i(k) \) expires after the \( k \)-th firing occurs in the system, the node \( i \) transmits the \( (k + 1) \)-th message, and is reset with \( \phi_i(k + 1) = 1 \), and \( r_i(k + 1) = T \).

### 3.2 Firing Message Transmission Time Interval of a Sensor Node

To analyze the transmission time interval of a sensor node, we consider the case where \( n \) nodes are adjacent to each other. This set of nodes is denoted by \( N \). We also denote the time that a sensor node \( i \) fires the \( m \)-th message in the system by \( t_m \). We assume that node \( i \) is firing \( x \) times until \( t_m \). Because node \( i \) fired at \( t_m \), it results in \( \phi_i(m) = 1 \), and \( r_i(m) = T \). Furthermore, the remaining nodes receiving the firing message from node \( i \) update their residual time, according to their phase angle.

Let \( j \) be the node with the longest residual time among the remaining nodes, except for node \( i \) in \( t_m \) (i.e., \( j = \arg\max_{a \in N - \{i\}} r_a(m) \)); \( j \) will receive \( (n-1) \) firing signals from other nodes before it fires \( (x + 1) \) times when \( r_j(m) < T \). Because \( r_i(k) \) increases with the phase angle, whenever node \( i \) receives the firing signal from other nodes, \( r_i(k) \) changes as shown in Figure 3, until \( i \) performs the next firing after \( t_m \).

Thus, the time taken for \( i \) to perform the \( (x + 1) \)-th firing is given by

\[
T + \sum_{a=1}^{n-1} (1 - \alpha)^a T = \frac{(1 - (1 - \alpha)^n)T}{\alpha}.
\]

### 4. Performance Evaluation

This section presents the analysis of the performance of the proposed technique through a simulation. A sink node was installed at the center of one cell, and then 10, 20, and 50 sensor nodes \((n)\) were randomly arranged in the cell. The initial phase angle \( \phi_i(0) \), of each sensor node, was randomly set between 0 and 1. Furthermore, by adjusting the transmission range of the nodes, we made two experimental scenarios. In a single cluster scenario, all the nodes are within the communication range of each other. In two cluster scenario, sensors are divided into two groups in order for the nodes in the same group to be adjacent to each other and the nodes in a different group act as a hidden node to the nodes in the other group because they are not within the transmission range of any other nodes in the other group. In all the experiments, the initial \( T \) value was set to 1 s, and \( \alpha \) as 0, 0.1, 0.4, 0.7, 1.0, and 1/\( n \). In this experiment, the node configuration was assumed to be the Tmote Sky mote \([15]\) based on the IEEE 802.15.4. Specifically, a node can send 250 kbps in a 2.4-GHz band. The transmission range of each node is set to 50 m. By setting the sink node such that it would not send any messages during the experiment, its effect on the performance measurement of the proposed technique was eliminated. The experimental time was set such that the ratio of the total number of firings to the number of sensor nodes was 20.

The performance evaluation factors included: inter-firing time for all and each sensor node, and data collision rate. Furthermore, we showed the fairness index. The inter-firing time for all sensor nodes refers to an average value of time intervals for all data transmissions occurring in the cell. The inter-firing time for each sensor node refers to the average of the transmission time interval values of each node. The data collision rate refers to the measured number of collisions in the data packet transmitted from the sensors due to the increase in the number of sensor nodes transmitting at a similar timing, or due to the increase in the number of hidden nodes divided by the total

Figure 3. Dynamics of \( r_i(k) \) from when node \( i \) fires \( x \) times to when node \( i \) fires \((x + 1)\) times.
number of data packet transmissions. Finally, the fairness was calculated as per Jain’s fairness index [16] in terms of the number of data transmissions, as follows:

\[
\text{fairness index} = \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n \cdot \sum_{i=1}^{n} x_i^2}.
\] (2)

In this study, we compare the performance of the proposed method with that of DESYNC method [11]. The DESYNC technique is a desynchronization-based sensor data transmission technique that uses the PCO model. In this technique, each node transmits its own phase angle \(\phi_i(k)\) information within the transmission range during firing. Every node performs firing once within the transmission interval of \(T\). One node adjusts its ensuing phase angle \(\phi_i(k+1)\) for the next firing, using the difference between \(\phi_{i-1}(k)\) and \(\phi_{i+1}(k)\), which are the phase angles of the nodes immediately before and after the firing node, respectively. The phase angle can be expressed as

\[
\phi_i(k+1) = (1 - \alpha)\phi_i(k) + \alpha \cdot \text{mid}(\phi_{i-1}(k), \phi_{i+1}(k)).
\] (3)

Figure 4 shows the inter-firing time for all sensor nodes, according to the change of \(\alpha\) values in two scenarios, namely a single cluster and two clusters. In Figure 4, in the DESYNC technique, because all sensor nodes transmit data once during a fixed transmission period \(T\), the inter-firing time for all sensor nodes is determined by \(1/n\). Furthermore, the change caused by \(\alpha\) values is relatively insignificant. Conversely, the proposed technique appropriately adjusts the transmission period of each sensor node to control the amount of redundant data. Moreover, because the performance change due to the \(\alpha\) values is large when the proposed method is used, \(\alpha\) setting is required for optimizing the performance. Thus, this study proposes a technique in which each sensor node can reliably transmit at least one data packet for every data transmission interval \(T\), required by the application service as a major design requirement. In this case, the meaning of reliably transmitting at least one data packet can be clarified by using the following example: if the bit error rate (BER) is \(10^{-3}\), and the packet length is 127 bytes (based on Tmote Sky mote), the probability that the sink node successfully receives the packet transmitted from the sensor node is approximately 0.36, 0.59, and 0.74 for one, two, and three transmissions, respectively, during \(T\). Using the proposed technique, assuming that an appropriate \(\alpha\) value is set, considering the channel transmission environment, at least one data packet can be reliably transmitted for each data transmission interval \(T\), while reducing the amount of redundant data.

Table 1 shows the inter-firing time for each sensor node, according to the change in \(\alpha\) values, in two experimental scenarios. The proposed technique controls the amount of redundant data within the cell by prolonging its data transmission period when firing occurs within the data transmission period \(T\). Thus, the inter-firing time for each sensor node is larger than \(T = 1\), particularly when \(\alpha = 0\). Here, the data transmission time interval of each node increases because a sensor node obtains a new \(T\) value by adding all the previous \(T\) values whenever it receives
Table 1. Comparison of inter-firing time for each sensor node

|                | Proposed technique | DESYNC       |
|----------------|--------------------|--------------|
|                | \(\alpha = 10\)    | \(\alpha = 20\) | \(\alpha = 50\) | \(\alpha = 10\)    | \(\alpha = 20\) | \(\alpha = 50\) |
|                | 1                  | 0.91         | 0.95          | 0.95          | 0.91         | 0.95          | 0.98          |
|                | 0.7                | 0.31         | 1.36          | 1.4           | 0.91         | 0.95          | 0.98          |
|                | 0.4                | 2.26         | 2.38          | 2.45          | 0.91         | 0.95          | 0.98          |
|                | 0.1                | 5.92         | 8.37          | 9.75          | 0.91         | 0.95          | 0.98          |
|                | \(1/n\)           | 6.14         | 8.37          | 9.75          | 0.91         | 0.95          | 0.98          |
|                | 0                  | 74.15        | 214.77        | 815.59        | 0.55         | 0.65          | 0.85          |
| Two clusters   | 1                  | 0.6          | 0.67          | 0.79          | 0.91         | 0.95          | 0.98          |
|                | 0.7                | 0.92         | 1.02          | 1.2           | 0.91         | 0.95          | 0.98          |
|                | 0.4                | 1.55         | 1.82          | 2.17          | 0.91         | 0.95          | 0.98          |
|                | 0.1                | 3.66         | 5.69          | 7.74          | 0.91         | 0.95          | 0.98          |
|                | \(1/n\)           | 3.86         | 10.18         | 26.66         | 0.91         | 0.95          | 0.98          |
|                | 0                  | 49.95        | 129.55        | 446.07        | 0.55         | 0.65          | 0.85          |

An inter-firing from other nodes. However, the DESYNC technique does not control the transmission amount of redundant data in the cell; thus, all the nodes have a fixed transmission period, and the transmission is set to occur once during the transmission period. Thus, the inter-firing time for each sensor node is approximately \(T = 1\).

The proposed technique aimed to solve the problem that data collisions increase due to the transmission of more data than is necessary from the sensor nodes. Figure 5 shows the data collision rates measured in the two experimental scenarios. Because the proposed technique adjusts both its transmission period and phase angle using the firing of adjacent nodes within the communication radius, it shows a higher performance than that of the DESYNC technique, which controls the phase angle using only the firing information of the two adjacent nodes.

Table 2 shows the fairness indices according to the change in \(\alpha\) values in the two scenarios. The significance of the fairness index lies in showing the difference between the numbers of firing times of the sensor nodes. As seen from the table, the DESYNC technique has a high fairness index because all the sensor nodes perform transmission once during the fixed transmission period \(T\), regardless of the transmission state in the cell. For the proposed method, each sensor node has a dynamic transmission period to control both data collision and redundant data transmission. However, Table 2 shows that the fairness index of the proposed technique is high for most \(\alpha\) values.

In summary, in the proposed technique each sensor node could control both unnecessary redundant data transmission and data collision by dynamically adjusting both the transmission period and phase angle, based on the firing of other nodes within the communication radius. Contrarily, in the DESYNC technique, each sensor node could, to a certain extent, maintain the firing interval between them at a certain level by dynamically adjusting only the phase angle, and only based on the firing information of the two adjacent nodes. Additionally, the technique showed a limitation in controlling unnecessary redundant data transmission and data collision by adjusting the phase angle, based on the fixed transmission period and limited information.

5. Conclusion

This study examined an application that provides a cost-effective monitoring service by collecting information using sensor nodes and transmitting it to a remote server. In this environment, we attempted to satisfy the following three requirements by allowing each sensor node to autonomously determine the data transmission timing. First, we intended to eliminate the redundant data, which exceeds the amount of information required by the application service. Second, we strived to reliably transmit at least one data packet for each data transmission period, as required by the application service. Third, we intended to
Figure 5. Comparison of data collision rate. (a) Single-cluster scenario and (b) Two-cluster scenario.

Table 2. Comparison of fairness indices

|                | Proposed technique | DESYNC |
|----------------|-------------------|--------|
|                | α = 10             | α = 20 | α = 50 | α = 10 | α = 20 | α = 50 |
| Single cluster |                   |        |        |        |        |        |
| 1              | 1                  | 1      | 1      | 1      | 1      | 1      |
| 0.7            | 1                  | 1      | 1      | 1      | 1      | 1      |
| 0.4            | 1                  | 1      | 1      | 1      | 1      | 1      |
| 0.1            | 1                  | 1      | 1      | 1      | 1      | 1      |
| 1/n            | 1                  | 1      | 1      | 1      | 1      | 1      |
| 0              | 1                  | 1      | 1      | 1      | 1      | 1      |
| Two clusters   |                   |        |        |        |        |        |
| 1              | 1                  | 1      | 1      | 1      | 1      | 1      |
| 0.7            | 1                  | 1      | 1      | 1      | 1      | 1      |
| 0.4            | 1                  | 1      | 1      | 1      | 1      | 1      |
| 0.1            | 0.88               | 0.71   | 0.92   | 1      | 1      | 1      |
| 1/n            | 0.88               | 0.55   | 0.54   | 1      | 1      | 1      |
| 0              | 1                  | 1      | 1      | 1      | 1      | 1      |

ensure that additional message exchange between sensor nodes would not be required for controlling the data transmission timing. We conducted a performance comparison in terms of the inter-firing time in other techniques, which have a fixed transmission period; the inter-firing time for each sensor node; the data collision rate; and the fairness index of sensor nodes, to verify that the proposed technique can efficiently control unnecessary redundant data transmission and data collision. To maximize the performance of the proposed technique, further studies are required to investigate optimization techniques for setting the $\alpha$ value, which is an important parameter for updating the transmission period.

Conflict of Interest

No potential conflict of interest relevant to this article has been reported.

Acknowledgements

This work was supported by Research Resettlement Fund for the new faculty of Kwangwoon University in 2019 and by the
National Research Foundation of Korea (NRF) and funded by the Korea government (No. NRF-2018R1D1A1B07050893).

References

[1] A. Mecocci and A. Abrardo, “Monitoring architectural heritage by wireless sensors networks: San Gimignano - a case study,” Sensors, vol. 14, no. 1, pp. 770-778, 2014. https://doi.org/10.3390/s140100770

[2] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Walsh, “Deploying a wireless sensor network on an active volcano,” IEEE Internet Computing, vol. 10, no. 2, pp. 18-25, 2006. https://doi.org/10.1109/MIC.2006.26

[3] X. Lu, S. Wang, W. Li, P. Jiang, and C. Zhang, “Development of a WSN based real time energy monitoring platform for industrial applications,” in Proceedings of 2015 IEEE 19th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Calabria, Italy, 2015, pp. 337-342. https://doi.org/10.1109/CSCWD.2015.7230982

[4] A. B. Noel, A. Abdaoui, T. Elfouly, M. H. Ahmed, A. Badawy, and M. S. Shehata, “Structural health monitoring using wireless sensor networks: a comprehensive survey,” IEEE Communications Surveys and Tutorials, vol. 19, no. 3, pp. 1403-1423, 2017. https://doi.org/10.1109/COMST.2017.2691551

[5] Z. Congyi, J. L. Mission, S. H. Kim, Y. S. Youk, and H. J. Kim, “Application of principal components analysis method to wireless sensor network based structural monitoring systems,” International Journal of Fuzzy Logic and Intelligent Systems, vol. 8, no. 1, pp. 11-17, 2008. https://doi.org/10.5391/IJFIS.2008.8.1.011

[6] Y. Xiao, H. Gao, J. Ding, X. Wang, and X. Meng, “An outlier detection method and its application for earthen ruins data monitored by WSN,” in Proceedings of 2017 5th International Conference on Advanced Cloud and Big Data (CBD), Shanghai, China, 2017, pp. 414-419. https://doi.org/10.1109/CBD.2017.78

[7] G. Anastasi, M. Conti, and M. Di Francesco, “A comprehensive analysis of the MAC unreliability problem in IEEE 802.15.4 wireless sensor networks,” IEEE Transactions on Industrial Informatics, vol. 7, no. 1, pp. 52-65, 2011. https://doi.org/10.1109/TII.2010.2085440

[8] S. Sotheara, K. Aso, N. Aomi, and S. Shimamoto, “Effective data gathering and energy efficient communication protocol in wireless sensor networks employing UAV,” in Proceedings of 2014 IEEE Wireless Communications and Networking Conference (WCNC), Istanbul, Turkey, 2014, pp. 2342-2347. https://doi.org/10.1109/WCNC.2014.6952715

[9] Y. S. Youk, S. H. Kim, and Y. H. Joo, “Intelligent data reduction algorithm for sensor network based fault diagnostic system,” International Journal of Fuzzy Logic and Intelligent Systems, vol. 9, no. 4, pp. 301-308, 2009. https://doi.org/10.5391/IJFIS.2009.9.4.301

[10] D. De Guglielmo, G. Anastasi, and M. Conti, “A localized de-synchronization algorithm for periodic data reporting in IEEE 802.15.4 WSNs,” in Proceedings of 2012 IEEE Symposium on Computers and Communications (ISCC), Cappadocia, Turkey, 2012, pp. 605-610. https://doi.org/10.1109/ISCC.2012.6249363

[11] J. Degesys, I. Rose, A. Patel, and R. Nagpal, “DESYNC: self-organizing desynchronization and TDMA on wireless sensor networks,” in Proceedings of the 6th International Conference on Information Processing in Sensor Networks, Cambridge, MA, 2007, pp. 11-20. https://doi.org/10.1145/1236360.1236363

[12] R. E. Mirollo, and S. H. Strogatz, “Synchronization of pulse-coupled biological oscillators,” SIAM Journal on Applied Mathematics, vol. 50, no. 6, pp. 1645-1662, 1990. https://doi.org/10.1137/0150098

[13] A. Patel, J. Degesys, and R. Nagpal, “Desynchronization: the theory of self-organizing algorithms for round-robin scheduling,” in Proceedings of the 1st International Conference on Self-Adaptive and Self-Organizing Systems (SASO), Cambridge, MA, 2007, pp. 87-96. https://doi.org/10.1109/SASO.2007.17

[14] C. S. Peskin, Mathematical Aspects of Heart Physiology. New York, NY: Courant Institute of Mathematical Sciences, New York University, 1975.

[15] Moteiv Corporation, “Tmote Sky: ultra low power IEEE 802.15.4 compliant wireless sensor module,” 2006, Available http://www.crew-project.eu/sites/default/files/tmote-sky-datasheet.pdf
[16] C. Guo, M. Sheng, Y. Zhang, and X. Wang, “A Jain’s index perspective on $\alpha$-fairness resource allocation over slow fading channels,” *IEEE Communications Letters*, vol. 17, no. 4, pp. 705-708, 2013. [https://doi.org/10.1109/LCOMM.2013.021913.130025](https://doi.org/10.1109/LCOMM.2013.021913.130025)

Jaesung Park received his B.S. and M.S. degrees in electronic engineering, and his Ph.D. in Electrical and Electronic Engineering from Yonsei University, Seoul, Korea in 1995, 1997, and 2001, respectively. From 2001 to 2002, he worked at a research faculty at the Department of Computer Science and Engineering of the University of Minnesota at Twin Cities, under a scholarship provided by LG Electronics Korea, where he worked as a senior research engineer from 2002 to 2005. He was an associate professor at the University of Suwon from 2005 to 2019. Currently, he is a professor at the School of Information Convergence in the Kwangwoon University. His current research interests include the design, analysis, and evaluation of communication networks.
E-mail: jaesungpark@kw.ac.kr

Changyong Yoon received his B.S., M.S., and Ph.D. in Electrical and Electronic Engineering from Yonsei University, Seoul, Korea, in 1997, 1999, and 2010, respectively. He was a senior research engineer at LG Electronics Inc. and LG-Nortel, and he developed system software for the DVR and WCDMA from 1999 to 2006. From 2010 to 2012, he was a chief research engineer at LG Electronics Inc. and LG-Nortel, and he developed system software for the DVR and WCDMA from 1999 to 2006. From 2010 to 2012, he was a chief research engineer at LG Display and developed circuit and algorithms for touch systems. Since 2012, he has been a professor with the Department of Electrical Engineering, Suwon Science College, Hwaseong, Korea. His main research interests include intelligent transportation systems, pattern recognition, robot vision, and fuzzy application systems.
E-mail: cyyoon@ssc.ac.kr