Embedded IoT Systems: Network, Platform, and Software

Guest Editors: Jeongyeup Paek Omprakash Gnawali, Marcos A. M. Vieira, and Shuai Hao
Embedded IoT Systems: Network, Platform, and Software
Embedded IoT Systems: Network, Platform, and Software

Guest Editors: Jeongyeup Paek Omprakash Gnawali, Marcos A. M. Vieira, and Shuai Hao
Editorial Board

M. Anastassopoulos, UK
C. Agostino Ardagna, Italy
J. M. Barcelo-Ordinas, Spain
Alessandro Bazzi, Italy
Paolo Bellavista, Italy
Carlos T. Calafate, Spain
M. Calderon, Spain
Juan C. Cano, Spain
Salvatore Carta, Italy
Yuh-Shyan Chen, Taiwan
Massimo Condoluci, UK
Antonio de la Oliva, Spain
Jesus Fontecha, Spain
Jorge Garcia Duque, Spain
L. J. García Villalba, Spain
Michele Garetto, Italy
Romeo Giuliani, Italy
Javier Gozalvez, Spain
Francesco Gringoli, Italy
Peter Jung, Germany
Dik Lun Lee, Hong Kong
Sergio Mascetti, Italy
Elio Masciari, Italy
Maristella Matera, Italy
Franco Mazzenga, Italy
Eduardo Mena, Spain
Massimo Merro, Italy
Jose F. Monserrat, Spain
Francesco Palmieri, Italy
J. J. Pazos-Arias, Spain
Vicent Pla, Spain
Daniele Riboni, Italy
Pedro M. Ruiz, Spain
Michele Ruta, Italy
S. Sardellitti, Italy
Floriano Scioscia, Italy
Laurence T. Yang, Canada
Jinglan Zhang, Australia
Contents

Embedded IoT Systems: Network, Platform, and Software
Jeongyeup Paek, Omprakash Gnawali, Marcos A. M. Vieira, and Shuai Hao
Volume 2017, Article ID 5921523, 2 pages

Virtual MIMO Beamforming and Device Pairing Enabled by Device-to-Device Communications for Multidevice Networks
Yeonjin Jeong, Jooheum Yoon, Sang Hyun Lee, and Yun Hee Kim
Volume 2017, Article ID 7652902, 10 pages

Multitiered and Distributed WSAN for Cooperative Indoors Environment Management
Taewook Heo, Hyunhak Kim, Yoonmee Doh, Kwangsoo Kim, Jongjun Park, Naesoo Kim, JongArm Jun, and JeongGil Ko
Volume 2017, Article ID 6979178, 12 pages

Collecting Big Data from Automotive ECUs beyond the CAN Bandwidth for Fault Visualization
Jeong-Woo Lee, Ki-Yong Choi, and Jung-Won Lee
Volume 2017, Article ID 4395070, 13 pages

Efficient Thread Mapping for Heterogeneous Multicore IoT Systems
Thomas Mezmur Birhanu, Zhetao Li, Hiroo Sekiya, Nobuyoshi Komuro, and Young-June Choi
Volume 2017, Article ID 3021565, 8 pages

Massive MIMO Relay Systems with Multipair Wireless Information and Power Transfer
Hongwu Liu, Kyung Hi Chang, and Kyung Sup Kwak
Volume 2017, Article ID 1760187, 17 pages

A Novel Low-Cost Real-Time Power Measurement Platform for LoWPAN IoT Devices
Yang Liu, Yubing Wang, Weiwei Gao, Wuxiong Zhang, Hua Qian, and Yang Yang
Volume 2017, Article ID 8713873, 14 pages

Small-World and Scale-Free Network Models for IoT Systems
Insoo Sohn
Volume 2017, Article ID 6752048, 9 pages
With the rapid development of connected embedded devices being placed everywhere in our everyday life, the vision of the Internet of Things (IoT) is coming close to reality where billions of physical world devices have a digital presence on the Internet. However, despite the importance of IoT, there are still several technical challenges remaining before the next Internet revolution. This special issue has focused on technical challenges that can enable IoT. We call for manuscripts presenting and discussing the most recent advances in embedded systems for IoT. Until the deadline of the special issue, 17 manuscripts have been received worldwide. After the review process, 7 manuscripts have been accepted by this special issue. The accepted research manuscripts have focused on the energy efficiency and power transfer for IoT devices, device-to-device communication, multithreading for multi-core IoT platforms, and topology construction and scalability of IoT networks. Accepted manuscripts present the important research findings and these advances will contribute to the development of IoT.

Evaluation of power efficiency is important for low power wireless personal area network (LoWPAN) devices and applications in IoT. Conventional methods to evaluate the power efficiency of LoWPAN devices rely deeply on the accuracy of the testing equipment, which trades off high cost with limited accuracy. To tackle this challenge, a low cost, real-time power measurement platform called PTone is proposed, which can be used to detect the real-time power usage of LoWPAN devices and be able to determine the state of each module of device under test. Based on PTone, an abnormal status diagnosis mechanism is developed which can not only judge abnormal status, but also classify the abnormal status and locate the abnormality causing module accurately.

Power transfer wirelessly is another promising technology that can enable a massive number of wireless devices in IoT. For this purpose, destination-aided simultaneous wireless information and power transfer (SWIPT) is proposed for a decode-and-forward relay network, in which massive multiple-input multiple-output (MIMO) antennas are deployed at relay to assist communications among multiple source-destination pairs. During relaying, energy signals are emitted from multiple destinations when multiple sources are sending their information signals to relay. Analysis reveals that asymptotic harvested energy is independent of the fast fading effect of wireless channels; meanwhile, transmission powers of each source and destination can be scaled down inversely proportional to the number of relay antennas. To reduce energy leakage interference and multipair interference, zero-forcing processing and maximum ratio transmission are employed at relays. Fundamental trade-off between harvested energy and achievable sum rate is quantified, and it is shown that asymptotic sum rate is neither convex nor concave with respect to power splitting and destination transmission power. Thus, a one-dimensional embedded bisection algorithm can be used to jointly determine the optimal power splitting and destination transmission power.
which shows that destination-aided SWIPT are beneficial for harvesting energy and increasing the sum rate.

For the past decade, wireless sensor networks have focused primarily on data collection. As a result, the network topology for these systems was usually heavily centralized. However, for these networks to form a full IoT system, the introduction of proper actuation units and decision-making intelligence is inevitable. To that end, a new distributed wireless sensor and actuator network (DWSAN) system has been proposed. The DWSAN system architecture effectively combines both sensor and actuation hardware devices to a single network and manages this network so that the actuation decisions are made in a distributed manner and the topology of the network maintains a multitier architecture. Intensive sets of evaluations reveal that, compared to the centralized approaches that have been used in most wireless sensor network systems until now, when actuation units are introduced to the system, the DWSAN architecture reduces the transmission load of the network and the actuation decision-making latency by close to twofold and threefold, respectively. Furthermore, this benefit naturally leads to better scalability of the system, making it suitable for various sensing applications in different environments.

Device heterogeneity and device-to-device (D2D) communication are another important enabler for the Internet of Things. For the D2D communication, transmit and receive beamforming with the channel state information for virtual MIMO enabled by D2D receive cooperation can be used. Analysis on the sum rate achieved by a device pair has been made, and a distributed algorithm for device pairing to maximize the throughput of the multidevice network has been developed in this special issue. Furthermore, for heterogeneous multicore IoT systems, efficient thread mapping and scheduling mechanism primed for heterogeneously configured multicore systems had been proposed which considers CPU utilization for mapping running threads with the appropriate core that can potentially deliver the actual needed capacity.

Finally, topology management for IoT has been studied analytically. For successful realization of IoT, challenges such as heterogeneous connectivity, ubiquitous coverage, reduced network and device complexity, enhanced power saving, and enhanced resource management have to be solved, and all these challenges are heavily impacted by the IoT network topology consisting of a massive number of connected devices. Small-world networks and scale-free networks are important complex network models with a massive number of nodes and have been actively used to study the network topology of brain networks, social networks, and wireless networks. These models, also, have been applied to IoT networks to enhance synchronization, error tolerance, and more. However, due to the interdisciplinary nature of the network science, with heavy emphasis on graph theory, it is not easy to study the various tools provided by complex network models. Therefore, the concept of basic graph theory has to be used, including small-world networks and scale-free networks, to provide system models that can be easily implemented to be used as a powerful tool in solving various research problems related to IoT.

**Acknowledgments**

We would like to thank all the reviewers that have contributed their time and insight to this special issue.

*Jeongyeup Paek*

*Omprakash Gnawali*

*Marcos A. M. Vieira*

*Shuai Hao*
Virtual MIMO Beamforming and Device Pairing Enabled by Device-to-Device Communications for Multidevice Networks

Yeonjin Jeong,1 Jooheum Yoon,1 Sang Hyun Lee,2 and Yun Hee Kim1

1Department of Electronics and Radio Engineering, Kyung Hee University, Yongin, Gyeonggi-do 17104, Republic of Korea
2School of Electrical Engineering, Korea University, Seoul 02841, Republic of Korea

Correspondence should be addressed to Yun Hee Kim; yheckim@khu.ac.kr

Received 23 November 2016; Accepted 30 January 2017; Published 21 May 2017

Academic Editor: Jeongyeup Paek

Copyright © 2017 Yeonjin Jeong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We consider a multidevice network with asymmetric antenna configurations which supports not only communications between an access point and devices but also device-to-device (D2D) communications for the Internet of things. For the network, we propose the transmit and receive beamforming with the channel state information (CSI) for virtual multiple-input multiple-output (MIMO) enabled by D2D receive cooperation. We analyze the sum rate achieved by a device pair in the proposed method and identify the strategies to improve the sum rate of the device pair. We next present a distributed algorithm and its equivalent algorithm for device pairing to maximize the throughput of the multidevice network. Simulation results confirm the advantages of the transmit CSI and D2D cooperation as well as the validity of the distributive algorithm.

1. Introduction

The Internet of things (IoT) connecting billions of devices is anticipated to make our society smarter by collecting the remotely sensed data from the devices and actuating the devices autonomously based on the collected data [1, 2]. The networks for the IoT would connect the devices not only to an access point (AP) but also to neighboring devices through direct communication links, which are often observed in machine-to-machine (M2M) and device-to-device (D2D) communications [3–5]. In addition, the direct links are controlled by the network as a whole or in part to guarantee the quality-of-services.

This paper utilizes the direct links of M2M and D2D communications to assist the information transfer from an AP to the devices through cooperation. There exist various cooperation methods such as amplify-and-forward (AF), decode-and-forward (DF), compress-and-forward (CF), and their variations to enlarge the coverage area or to improve the reliability of information delivery [6–9]. Among the methods, CF-based receive cooperation has attracted attention recently as a method of improving the system capacity by forming a virtual multiple-input multiple-output (MIMO) configuration with single-antenna devices [10–13].

The CF-based receive cooperation is considered for an AP with two antennas supporting two devices in [10, 11] from the aspect of physical layer designs with adaptive modulation and coding. A higher layer issue of device matching and resource sharing has been studied for the network with multiple devices in [12, 13], where a distributive message passing algorithm is derived based on the bargaining in exchange networks [14–18]. These studies have shown that the CF-based receive cooperation not only enjoys a proximity gain of the closely located devices but also provides a flexibility in the design of the D2D links by allowing any type of physical layer standard for the D2D links.

This paper proposes a network-assisted D2D communication network as one wireless communication platform for the IoT, which forms the virtual MIMO with multiple-antenna devices to enhance the rate from the physical layer aspect and implements the distributed device pairing algorithms to improve the overall network throughput. The contributions of this paper are summarized as follows. Firstly, the beamforming (BF) and power allocation (PA) strategies exploiting the transmit CSI are proposed for the virtual MIMO to improve the rate performance of the downlink physical layer. Secondly, several distributive algorithms for device pairing are derived from the bargaining [15] and
auction [19] to provide the network throughput equivalent to that of the optimal pairing. Thirdly, we verify the synergy of combining the virtual MIMO and device pairing algorithms under various system configurations in the scenario of IoT with D2D communications. It should be further noted that the D2D-assisted virtual MIMO with its enhancements can be adopted in the mobile cellular networks providing the proximity services [4] and the distributive pairing algorithms can be combined with various cooperation strategies among the devices for information sharing and processing [9].

The rest of this paper is organized as follows. After providing notations used in this paper, we describe the system model with and without D2D cooperation in Section 2. Section 3 investigates several methods to improve the sum rate of the two devices in D2D cooperation. Device pairing algorithms are devised from bargaining and auction to maximize the network throughput in Section 4. The performance of the network is evaluated in different aspects in Section 5 followed by the concluding remarks in Section 6.

**Notations.** We denote by $A^T$, $A^H$, $\text{tr}[A]$, and $[A]_{ij}$ the transpose, Hermitian, trace, and $(k,l)$th element of a matrix $A$, respectively; $\text{diag}(x)$ denotes the diagonal matrix with diagonal vector $x$, $I_N$ denotes the $N \times N$ identity matrix, and $0_{N \times M}$ denotes the $N \times M$ matrix with all-zero elements. We denote by $C^{\text{mst}}$ the space for $m \times n$ complex matrices. The operator $E[\cdot]$ denotes the expectation, $\sim$ signifies “distributed as,” and $\mathcal{C}\mathcal{M}(\mathbf{x}, \Sigma)$ denotes the distribution of a vector $\mathbf{x}$ with complex Gaussian elements with mean $\mathbf{x}$ and covariance matrix $\Sigma$.

### 2. System Model

Consider an IoT network with an AP and $K$ devices as described in Figure 1(a), where the AP is equipped with $N_T$ transmit antennas and each device is equipped with $N_R$ receive antennas. We consider asymmetric antenna configurations as $N_T \geq 2N_R$, which are often encountered in practical scenarios with high complexity APs and low complexity IoT devices. The downlink (DL) from the AP to the devices utilizes a licensed spectrum of bandwidth $B_C$ for long-range communications, while the D2D links utilize an unlicensed spectrum of bandwidth $B_D$ for relatively short-range communications as shown in Figure 1(b); although an unlicensed spectrum is also available at a very high frequency as 60 GHz, we limit our description to an unlicensed spectrum at a carrier frequency below 6 GHz which has a propagation model similar to that of the licensed spectrum. The DL frame is divided into $K$ equilength time slots, each of which is dedicated to each device to be shared with the paired device.

The channel from the AP to device $k$ is modeled by the channel response matrix $H_k = [h_{k,1}, h_{k,2}, \ldots, h_{k,N_R}]^T \in C^{N_R \times N_T}$. The component vector is modeled as $h_{k,j} \sim \mathcal{C}\mathcal{N}(0_{N_T \times 1}, \omega_k \chi_k I_{N_T})$, where $\omega_k = \omega_0 d_k^{-\nu}$ is the path-loss at device $k$ determined by the distance $d_k$ from the AP, path-loss constant $\omega_0$, and path-loss exponent $\nu$, and $\chi_k$ is the log-normal random variable reflecting the shadowing.

![Figure 1: System model: (a) network configuration; (b) frequency bands.](image)

The D2D link from device $i$ to device $j$ is described by the channel matrix $G_{ij} = [g_{i,j,1}, g_{i,j,2}, \ldots, g_{i,j,N_R}]$, where $g_{i,j,k} \sim \mathcal{C}\mathcal{M}(0_{N_R \times 1}, \omega_{ij} \chi_{ij} I_{N_R})$ and $\omega_{ij} = \omega_0 d_{ij}^{-\nu}$ is the path-loss between devices $i$ and $j$ at distance $d_{ij}$. The channel reciprocity holds in D2D links as $G_{ij} = G_{ji}$. This paper assumes that the AP has the perfect CSI on the channels $\{H_k\}_{k=1}^K$. The network pairs the devices for signal transmission with transmit beamforming (BF) over the time slots allocated to the devices. Without D2D receive cooperation, the network supports $N_T \times N_R$ MIMO from the AP to each device in the pair. If the devices in the pair participate in D2D receive cooperation, the network forms $N_T \times 2N_R$ virtual MIMO from the AP and each device in the pair which can support $\min(N_T, 2N_R)$ spatially multiplexed symbols.

Let devices $i$ and $j$ be paired for signal transmission over their time slots with transmit BF $W_{ij}$. The transmit symbol vector for devices $(i, j)$ is modeled as

$$x_{ij} = \sqrt{\theta P_A} W_{ij} s_{ij},$$

where $P_A$ is the transmit power of the AP, $s_{ij}$ are the $2N_R \times 1$ symbol vector with $E[s_{ij}^H s_{ij}] = (1/2N_R)I_{2N_R}$, and $\theta = 2N_R/\text{tr}[W_{ij} W_{ij}^H]$ is the scaling factor satisfying the transmit power $P_A$. The corresponding received signal vector at device $k \in \{i, j\}$ is expressed as

$$y_k = [y_{k,1}, y_{k,2}, \ldots, y_{k,N_R}]^T = \sqrt{\theta P_A} H_k W_{ij} s_{ij} + n_k,$$

where $n_k \sim \mathcal{C}\mathcal{M}(0_{N_R \times 1}, \sigma^2 I_{N_R})$ is the background-noise vector at device $k$. 


With D2D cooperation that exchanges the compressed versions of the received signals between the paired devices, the received signals available at device $k \in \{i, j\}$ are given by

$$\tilde{y}_{ij} = \left[y_{ij}^T, \tilde{y}_{ij}^T\right]^T = \sqrt{\theta P_A} H_{ij} W_{ij} s_{ij} + \tilde{n}_{ij},$$

(3)

where $H_{ij} = [H_{ij}^T, H_{ij}^T]^T$ is the channel matrix of the pair and $\tilde{n}_{ij} = [\tilde{n}_{ij}^T, \tilde{n}_{ij}^T]^T$ with $\tilde{n}_{ij} = n_{ij} + n_{Q,j}$. Here, $n_{Q,j} = [n_{Q,j,1}, n_{Q,j,2}, \ldots, n_{Q,j,N_q}]^T$ is a vector of noises that the compression in D2D cooperation incurs. With Voronoi vector quantization, the compression noises can be modeled as zero-mean complex Gaussian with $E[n_{Q,j}, n_{Q,j}^H] = \text{diag}(\sigma^2_Q)$ [10], where $\sigma^2_Q = [\sigma^2_{Q,j,1}, \sigma^2_{Q,j,2}, \ldots, \sigma^2_{Q,j,N_q}]^T$ and $\sigma^2_{Q,j,m}$ is bounded from the rate-distortion theory [20] as

$$\sigma^2_{Q,j,m} = \frac{E[|y_{j,m}|^2 h_{j,m}]}{2^{r_{Q,j,m}} - 1}$$

(4)

where $r_{Q,j}$ is the source coding rate employed at device $j$ for compression of $\{y_{j,m}\}_{m=1}^{N_q}$. Therefore, we have

$$R_{n_j} = E[\tilde{n}_{ij}^H \tilde{n}_{ij}] = \begin{bmatrix} \sigma^2_{Q,j,1} & 0_{N_x \times N_q} \\ 0_{N_x \times N_q} & \text{diag} (\sigma^2_{Q,j}) \end{bmatrix}.$$  

(5)

Under this model, the network employs the transmit BF at the BS; that is,

$$W_{ij,2d} = \tilde{V}_{ij}$$

(6)

and $\theta_{2d} = 1$, where $\tilde{V}_{ij} \in C^{2N_x \times 2N_x}$ is the matrix constructed with the first $2N_x$ columns of $V_{ij}$ from singular value decomposition (SVD) of the pair channel matrix $H_{ij} = U_{ij} \Sigma_{ij} V_{ij}^H$. Device $i$ receiving signals in (3) through D2D cooperation uses the receive BF $U_{ij}^H$ such that

$$z_{ij} = U_{ij}^H \tilde{y}_{ij} = \sqrt{P_A} A_{ij} s_{ij} + U_{ij}^H \tilde{n}_{ij}.$$  

(7)

With D2D cooperation, the transmit symbol vector $s_{ij}$ can be devoted to symbol transmission, for device $i$ only, for device $j$ only, or for both, according to the resource allocation.

For comparison, we consider the system without D2D cooperation. Up to $N_R$ symbols can be multiplexed for each device at the AP as $s_k = [s_k^I, s_k^H]^T$, where $s_k \in C^{N_x \times 1}$ is the symbol vector for device $k \in \{i, j\}$. The transmit BF $W_{ij,\text{no}}$ without D2D cooperation can be designed as

$$W_{ij,\text{no}} = [W_{i,\text{no}}, W_{j,\text{no}}]^T,$$

(8)

by means of the block diagonalization (BD) method to decode $s_k$ from (2) at device $k \in \{i, j\}$ as in [21]. To be specific, $W_{i,\text{no}}$ is designed to be orthogonal to $H_i$ but to be steered to $H_i$ as $W_{i,\text{no}} = V_i^* V_i, \text{eq}$, where $V_i \in C^{N_y \times (N_y - N_R)}$ is the submatrix of $V_i$ corresponding to the null space of $H_i = U_{ij} \Sigma_{ij} V_{ij}^H$ and $V_{i,\text{no}} \in C^{N_y - N_R \times N_y}$ is the submatrix corresponding to nonzero eigenvalues from the right unitary matrix $V_{i,\text{no}}$ of $H_i V_i^H = U_{i,\text{no}} \Sigma_{i,\text{no}} V_{i,\text{no}}^H$. The design of $W_{ij,\text{no}}$ is obtained in a similar way. Therefore, we have

$$W_{ij,\text{no}} = [V_j^H V_{i,\text{bd}}, V_j^H V_{j,\text{bd}}]$$

(9)

and $\theta_{\text{no}} = 1$. By applying $U_{k,\text{bd}}^H$ to $y_k$ at device $k \in \{i, j\}$, we have

$$z_k = U_{k,\text{bd}} y_k = \sqrt{P_A} \Sigma_{k,\text{bd}} s_k + U_{k,\text{bd}} n_k.$$  

(10)

It should be noted that $2N_R$ received signals are available at each device, while $N_R$ received signals are available at each device, and thus BF strategies are different.

### 3. Improvement of Sum Rate of a Device Pair

With D2D cooperation, the sum rate achieved by the pair $(i, j)$ is a linear combination of rates $R_{ij}$, delivered to device $i$ when the two time slots are devoted to device $i$, and $R_{ji}$, delivered to device $j$ when the two time slots are devoted to device $j$. The sum rate achieved by a device pair is given by

$$\delta_{ij} = \kappa_i R_{ij} + \kappa_j R_{ji} \text{ bps},$$

(11)

where nonnegative factors $\kappa_i$ and $\kappa_j$ denote the fractions of the two time slots allocated to devices $i$ and $j$, respectively, subject to $\kappa_i + \kappa_j = 1$. Here, $R_{ij}$ is given by

$$R_{ij} = \frac{2B_C}{K} \sum_{m=1}^{2N_x} \log_2 \left(1 + \frac{P_A}{2N_R \sigma^2_{ij,m}} \lambda_{ij,m} \right) \text{ bps}$$

(12)

from (7), where $\lambda_{ij,m} = |\Sigma_{ij,m}|$ and

$$\sigma^2_{ij,m} = |U_{ij}^H R_{ij} U_{ij}|_{mm}.$$  

(13)

The rate $R_{ji}$ delivered to device $j$ is also obtained in a similar form by exchanging $j$ and $i$.

To improve the rate (12), we need to reduce $\sigma^2_{ij,m}$ or equivalently $\sigma^2_{Q,j,m}$ given in (4). We can reduce $\sigma^2_{Q,j,m}$ by increasing the source coding rate $r_{Q,j}$ as much as possible. The source coding rate for device $j$ to compress the received signal samples $\{y_{j,m}\}_{m=1}^{N_q}$ to be forwarded to device $i$ is limited by the channel capacity

$$C_{ij} = B_D \log_2 \text{det} \left[I_{N_x} + \frac{P_D}{N_R \sigma^2_{ij}} G_{ij}^H G_{ij} \right]$$

(14)
of the D2D link, where \( P_D \) and \( \sigma_D^2 \) are the transmit power and noise power of the device, respectively. Specifically, for the forwarded samples to be recovered at device \( i \) without errors, their transmission rate \( B_C N_p r_{Q_{ij}} \) over the D2D link should not exceed \( C_{ij} \). Therefore, the source coding rate \( r_{Q_{ij}} \) is upper-bounded by

\[
r_{Q_{ij}}^{\text{max}} = \frac{\eta}{N_R} \log_2 \det \left( I_{N_R} + \frac{P_D}{N_R \sigma_D^2} G_j G_j^H \right),
\]

(15)

where \( \eta = B_D/B_C \) is the bandwidth ratio.

With the maximum source coding rate, the variance of the compression noise is given by

\[
\sigma_{Q_{ij,m}}^2 \leq \frac{\sigma^2 \left( 1 + \left( \frac{P_D}{2N_R \sigma^2} \right) \| h_{i,m} \|^2 \right)}{\det \left( I_{N_R} + \frac{P_D}{N_R \sigma_D^2} G_j G_j^H \right)^{1/N_R} - 1}
\]

(16)

from (4) and (15). From (16), we can suggest some strategies to improve the sum rate achieved by a device pair in D2D cooperation. Firstly, the compression noise variance decreases exponentially as the bandwidth expansion \( \eta \) increases. For the fixed bandwidth expansion ratio as \( \eta = 1 \), the compression noise variance is reduced when the D2D link quality is much better than the DL quality in an average sense as

\[
\frac{P_A}{\sigma^2} \omega_{ij} < \frac{P_D}{\sigma_D^2} \omega_{ij}.
\]

(17)

We further improve the sum rate of the device pair by applying power allocation (PA) \( \psi_{ij} = [\psi_{i,1}, \psi_{i,2}, \ldots, \psi_{i,2N_R}]^T \) such that \( E_s[\psi_{ij} H_i^H] = \text{diag}(\psi_{ij}) \) and \( \sum_{m=1}^{2N_R} \psi_{i,m} = 1 \). In the case, the sum rate of the device pair is given by

\[
\mathcal{R}_{ij} = \frac{2B_C}{K} \sum_{m=1}^{2N_R} \log_2 \left( 1 + \frac{P_A}{\sigma^2} \lambda_{i,m} \psi_{i,m} \right) \text{bps},
\]

(18)

which is maximized by applying the water-filling algorithm with the information on \( \{(P_A/\sigma^2 \lambda_{i,m}} \}_{m=1}^{M} \) available at the AP; the noise variance \( \sigma_{j,m}^2 \) is not easy to be informed to the AP since it depends on the transmit signal and D2D link qualities. Therefore, the AP can compute the PA with water-filling algorithm by assuming \( \sigma_{j,m}^2 = \sigma^2 \). This sum rate of the device pair becomes the maximum when \( \sigma_{j,m}^2 = \sigma^2 \) or equivalently when \( \eta = \infty \).

For comparison, the achievable sum rate of pair \( (i,j) \) using two slots without D2D cooperation is given by \( S_{i,n}=R_{i,n} + R_{j,n} \), where

\[
R_{k,n} = \frac{2B_C}{K} \sum_{m=1}^{N_R} \log_2 \left( 1 + \frac{P_A}{2N_R \sigma^2} \lambda_{k,m,bd} \right) \text{bps}
\]

(19)

from (10), where \( \lambda_{k,m,bd} = [\Sigma_{k,bd}^2]_{m,n} \).

### 4. Distributed Pairing Algorithms

This section aims at forming the pairs to maximize the average normalized throughput of the network. The problem is formulated as

\[
\max_{\Pi} T_{\text{avg}} = \frac{1}{K} \sum_{i=1}^{K} \sum_{j=i+1}^{K} E \left\{ \delta_{ij} r_{ij} \right\} \text{bps/Hz},
\]

(20a)

s.t.

\[
\sum_{i=1}^{K} \pi_{ij} = 1,
\]

(20b)

\[
\sum_{j=1}^{K} \pi_{ij} = 1,
\]

(20c)

\[
\pi_{ij} \in \{0,1\},
\]

\[
\forall i,j,
\]

where \( \Pi \) is the \( K \times K \) matrix with \([\Pi]_{i,j} = \pi_{ij} \). The pairing indicator \( \pi_{ij} \in \{0,1\} \) is given by \( \pi_{ij} = 1 \) when devices \( i \) and \( j \) are paired and \( \pi_{ij} = 0 \), otherwise.

We first solve the problem by using a distributed algorithm based on Nash bargaining as proposed in [12] not only to maximize the average throughput but also to satisfy each device's desire to obtain the best bargaining gain. We then derive the auction algorithms for device pairing which are equivalent to the bargaining algorithm.

#### 4.1. Bargaining Algorithm Based on Message Passing

The problem is described by a graph \((\mathcal{V}', \mathcal{E})\) in an exchange network [14], where \( \mathcal{V}' \) is the set of vertices corresponding to the devices and \( \mathcal{E} \) is the set of edges corresponding to the D2D links among the devices. The edge \((i,j) \in \mathcal{E} \) connecting vertices \( i \) and \( j \) is associated with a positive weight

\[
\omega_{ij} = \delta_{ij} \min \left( \mathcal{R}_{ij}, \mathcal{R}_{ji} \right)
\]

(21)

which corresponds to the sum rate of the device pair \((i,j)\) ensured by any sharing policy \((\kappa_i, \kappa_j)\); without knowing \((\kappa_i, \kappa_j)\) in advance, we estimate the sum rate of the device pair pessimistically with the minimum value.

Let \((\mathcal{M}(\Pi), \xi)\) denote a configuration of the device pairing and resource sharing, where \(\mathcal{M}(\Pi)\) is a matching described by \(\Pi\) in graph \((\mathcal{V}', \mathcal{E})\) and \(\xi = \{\xi_i : i \in \mathcal{V}'\}\) is the set of the rates that the devices can obtain from the matching \(\mathcal{M}(\Pi)\). The solution is obtained with a balanced configuration [14] satisfying the stability

\[
\xi_i + \xi_j = \omega_{ij}
\]

(22)

for device pair \((i,j)\) contained in the matching \(\mathcal{M}(\Pi)\) and the balance

\[
\xi_i - \max_{k \in \mathcal{N}(i)} \left( w_{ik} - \xi_k \right) = \xi_j - \max_{l \in \mathcal{N}(j)} \left( w_{lj} - \xi_l \right)
\]

(23)

for all \((i,j) \in \mathcal{M}(\Pi)\), where \(\mathcal{N}(i)\) is the set of neighbors of device \(i\).
Algorithm 1: Distributed matching and resource sharing algorithm.

(1) Set $\mu_{ij}^{(0)} \leftarrow 0$ for $(i, j) \in \mathcal{E}$.
(2) repeat
    (3) At device $i$, update offer $\phi_{ij}^{(t)}$ using (24) for $j \in \mathcal{N}(i)$ and send it back to device $j$.
    (4) On receipt of $\phi_{ij}^{(t)}$ for all devices, update $\zeta_{ij}^{(t)}$ using (25).
    (5) At device $i$, update alternative rate $\mu_{ij}^{(t+1)}$ using (26) for $j \in \mathcal{N}(i)$ and send it back to device $j$.
    (6) $t \leftarrow t + 1$.
(7) until All messages converge or max number of iterations are reached.
(8) At device $i$, fix $\mu_{ij}$ and $\phi_{ij}$ for all $j \in \mathcal{N}(i)$ to determine $\zeta_i$ using (25), and find partner device $k \in \mathcal{N}(i)$ such that $\zeta_i = \phi_{ki}$.

This stable balanced configuration can be found by a distributed algorithm based on a message passing (MP) framework [17, 18, 22, 23], where each device transfers a nonnegative message on the current best alternative in device matching to its neighbors at each time instant. Let $\mu_{ij}^{(t)}$ denote the message from device $i$ to device $j$ at time $t$, representing the resource share which is guaranteed for device $i$ in the pair $(i, j)$. That is, $\mu_{ij}^{(t)}$ is an alternative rate that device $i$ can obtain maximally up to time instant $t$ when pairing with its neighbor $k \in \mathcal{N}(i) \setminus j$ except for $j$. Initially, $\mu_{ij}^{(0)}$ is set to zero for all $(i, j) \in \mathcal{E}$ since there is no guarantee ensured by the neighboring devices initially. Based on the alternative rate, an individual device makes an offer $\phi_{ij}^{(t)}$ to each of its neighbors for D2D cooperation, where the offer from device $i$ to device $j$ is computed as

$$\phi_{ij}^{(t)} = \left( w_{ij} - \mu_{ij}^{(t)} \right)_+ + \frac{1}{2} \left( w_{ij} - \mu_{ij}^{(t)} - \mu_{ji}^{(t)} \right)_+. \quad (24)$$

With the offers from the other devices, device $i$ estimates its maximal rate using

$$\zeta_i^{(t)} = \max_{k \in \mathcal{N}(i)} \phi_{ki}^{(t)}. \quad (25)$$

The update rule for $\mu_{ij}^{(t)}$ is given by

$$\mu_{ij}^{(t+1)} = (1 - \xi) \mu_{ij}^{(t)} + \xi \mu_{ij}^{(t+1)},$$

$$\mu_{ij}^{(t+1)} = \max_{k \in \mathcal{N}(i)} \phi_{ki}^{(t)}, \quad (26)$$

where $\mu_{ij}^{(t+1)}$ is an estimate of the rate guaranteed for device $i$ and $\xi = 0.7$ is the damping factor [12]. With the updated $\mu_{ij}^{(t+1)}$, the repeated updates of $\phi_{ij}^{(t+1)}$ and $\zeta_{ij}^{(t+1)}$ proceed using (24) and (25), respectively. The iteration continues until $\mu_{ij}^{(t+1)}$ converges for all available pairs $(i, j)$. Let $\bar{\mu}_{ij}, \bar{\phi}_{ij}$, and $\bar{\zeta}_i$ denote the corresponding values to which $\mu_{ij}^{(t)}, \phi_{ij}^{(t)}$, and $\zeta_i^{(t)}$ converge, that is, a fixed point of the algorithm. Upon convergence, device $i$ sets its share to $\bar{\zeta}_i$ and chooses device $k \in \mathcal{N}(i)$ with $\phi_{ki} = \bar{\zeta}_i$ as its partner. From $\bar{\zeta}_i$, the device obtains the resource sharing factor $\kappa_i = \bar{\zeta}_i / w_{ij}$. This procedure is summarized in Algorithm 1.

4.2. Equivalent Auction Algorithm for Device Pairing. The auction algorithm finds the maximum weighted matching (MWM) via an auction, based on the notion of buyers and prices [19]. The MWM corresponds to the optimal pairing providing the maximum throughput so that we briefly describe the auction algorithm in terms of messages defined in the previous section.

We can set the alternative rate $\mu_{ij}^{(t)}$ to the price which buyer $i$ pays to buy product $j$ and set the sum rate $w_{ij}$ to the value of product $j$, respectively. The benefit that can be earned when buyer $i$ buys product $j$ is given by $w_{ij} - \mu_{ij}^{(t)}$. The goal is to find the assignment of all buyers and products that maximizes the overall benefit, that is, $\sum_{i, j} (w_{ij} - \mu_{ij}^{(t)}) \pi_{ij}$. Therefore, the sum $\sum_{i, j} \mu_{ij}^{(t)} \pi_{ij}$ of the prices is constant and the MWM maximizes the overall benefit [15]. To handle this problem, the auction algorithm proceeds as in Algorithm 2. This algorithm is known to converge to the MWM if a solution exists [15, 19].

In this algorithm, the maximum is taken over all incoming $\phi_{ij}^{(t)}$ when updating $\mu_{ij}^{(t+1)}$. In addition, device $i$ does not bid at every iteration and to every neighboring device, thereby slowing down the convergence speed. Also, the auction algorithm works only for a bipartite network where one class of buyers is assigned only to the other class of products. Thus, the algorithm should be modified to allow the MWM for nonbipartite network where buyers can also become products. Note that, in a nonbipartite network, the net benefit of device $i$ corresponds to the price that device $j$ should pay to form a pair $(i, j)$. Subsequently, device $i$ does not necessarily consider the offer from device $j$ to determine the price that device $j$ should pay for pairing. To all of those objectives, the auction algorithm is modified to obtain a new algorithm running in a nonbipartite network, and the resulting algorithm is summarized in Algorithm 3.

In this algorithm, device $i$ can bid to all neighboring device $j$ at each iteration instead of updating $\phi_{ij}^{(t)}$ by adding a small constant $\delta$. Here, note that the auction algorithm in this subsection bears very close resemblance to the bargaining algorithm in Section 4.1. If we drop the second term in (24), which corresponds to dividing the surplus rate gain equally, we end up with Algorithm 3. Thus, the developed algorithm only differs from the modified auction algorithm in that it
includes an extra term that reduces “offers” by half the surplus rate gain. In fact, this algorithm is proved to converge to the MWM if a unique solution exists using the same technique as [12].

We now consider the generalization of the message update rule for (24) as

\[
\phi_{ij}^{(t)} = (w_{ij} - \mu_{ij}^{(t)})_+ + \gamma (w_{ij} - \mu_{ij}^{(t)} - \mu_{ji}^{(t)})_+.
\]

(27)

We see that (27) leads to the bargaining algorithm in Section 4.1 with \( \gamma = 1/2 \) and the modified auction algorithm with \( \gamma = 0 \), respectively. In fact, for \( \gamma \in [0,1) \), the corresponding algorithm finds the MWM if a unique solution exists. This statement can be proved using the technique in [16]. Thus, the distributive algorithms provided in this work can also find the MWM.

5. Numerical Results

We evaluate the performance of the network with \( K = 8 \) devices when \( P_A = 29 \text{ dBm}, P_D = 23 \text{ dBm}, B_C = 10 \text{ MHz}, \) and \( \sigma^2 = -104.5 \) dBm. The devices are clustered in the circle of radius \( r_o \) of which the center is located at distance \( d_o \) from the AP. In the circle, the location of the devices is randomly generated with uniform distribution. The SNR at the center of the cluster circle is denoted by \( \rho_o \) for a reference parameter. We assume the unlicensed spectrum of bandwidth \( B_D = 10 \text{ MHz}, 20 \text{ MHz}, \) and \( 40 \text{ MHz} \) for \( \eta = 1, \eta = 2, \) and \( \eta = 4 \). The path-loss exponent is set to \( \nu = 3.76 \) and the shadowing deviation is set to 10 dB for both the licensed and unlicensed spectrums.

Figure 2 shows the performance of the cluster with \( d_o = 300 \text{ m} \) and \( r_o = 200 \text{ m} \) as the cluster center SNR \( \rho_o \) varies when \( \eta = 2, N_{RP} = 2, N_{RS} = 1, \) and \( \sigma_o = \rho_o - 133.5 \) in dB. In the figure, “D2D-PA” and “D2D” indicate the proposed schemes with water-filling PA and equal PA, respectively, while “No D2D” denotes the BD-based BF without D2D cooperation and “D2D (No CSI)” indicates the D2D cooperation without the transmit CSI; here, “D2D (No CSI)” applies the zero-forcing receive BF at the devices. For the benchmark, we also provide the performance with ideal D2D link (“Ideal”) having no compression noise, or equivalently \( \eta = \infty \). The lines in the figure correspond to the results with optimal device matching obtained by finding the maximum value after searching for all possible pairing configurations while the marks correspond to the results with the distributive algorithm presented in the paper. The figure shows that the distributive algorithm provides the performance almost indistinguishable from the optimal performance. The transmit CSI on the DL improves the average throughput for both cases of the network with D2D cooperation and with no D2D cooperation. The best performance is achieved with “D2D-PA” exploiting the transmit CSI and D2D cooperation.

Figure 3 illustrates how the pairs are matched by the distributive bargaining algorithm for a single channel realization generated in Figure 2 when \( \rho_o = 10 \) dB. The left and right
Figure 2: Average normalized throughput of the cluster of radius $r_o = 200$ m located at $d_o = 300$ m as a function of the cluster center SNR $\rho_o$ when $\eta = 1$, $N_T = 4$, and $N_R = 2$.

Figure 3: A single realization of the device pairing and resource sharing when $\rho_o = 10$ dB in Figure 2: (a) D2D; (b) No D2D.

Subfigures describe a matching of "D2D" and "No D2D," respectively, for the same channel realization. The numbers inside the squares identify the devices and the numbers near to edges of the same color are the sum rates (weights) of the device pairs associated with the edges. We also show the rates (shares) of the devices near to the devices in Figure 3(a). Note that "D2D" can support the rates of the devices by choosing the resource sharing factors ($\kappa_i, \kappa_j$) while "No D2D" cannot support them. With D2D cooperation, the devices in proximity are likely to be paired to improve the quality of the D2D links as shown in Figure 3(a). On the other hand, the devices without D2D cooperation tend to be paired independent of their locations but dependent on the DL channel qualities as shown in Figure 3(b). As expected from the distributive algorithm, a device in a better condition tends to take higher rate than the other device in the pair with D2D cooperation; for instance, device 1 having more neighbors to bargain with tends to take higher rate than device 2 having less neighbors.

Figure 4 compares the performance of the network with that of transmit CSI for different antenna configurations $N_T \times N_R$ when $N_T = 2N_R$ and the other conditions remain unchanged from Figure 2. As the number $N_T$ ($N_R$) of antennas increases, the throughput is improved for all methods by multiplexing more symbols at the transmitter. In addition, the gain of "D2D-PA" over "No D2D" increases with the number of antennas by concentrating the signal power to the desired channel modes. Unfortunately, the performance gain with D2D cooperation is marginal for $2 \times 1$ antenna configuration.
Figure 4: Average normalized throughput of the cluster for several antenna configurations as the cluster center SNR $\rho_o$ varies.

Figure 5: Average normalized throughput of the cluster as a function of the cluster center SNR $\rho_o$ for several $\eta$ values when $N_T = 2$ and $N_R = 1$.

To improve the performance of D2D cooperation for $N_T = 2$ and $N_R = 1$, we increase the bandwidth expansion ratio $\eta$ in Figure 5 while the other conditions remain unchanged from Figure 2. The D2D cooperation methods are benefited by improving the D2D link qualities through increasing the D2D bandwidth irrespective of the transmit CSI available at the AP or not. The gain is more prominent when $\eta$ increases from 1 to 2.

Figure 6 compares the performances as the cluster location $d_o$ varies when $\eta = 1$ and $r_o = 25$ m. The antenna configuration is set to $N_T = 2$ and $N_R = 1$ in Figure 6(a) and $N_T = 4$ and $N_R = 2$ in Figure 6(b). The cluster center SNR is 0 dB at $d_o = d_{o,\text{max}} = 300$ m for both figures and the cluster center SNR at distance $d_o$ is obtained from $\rho_o = (d_{o,\text{max}}/d_o)^\eta$. The results reveal that the gain of D2D cooperation is larger when the cluster location gets farther from the AP since the quality of the D2D links relative to the DL gets better. This observation comes from the fact that the signals with larger amplitude suffer from larger quantization errors for the fixed number of quantization levels. The figure also shows that the effect of the compression noises becomes negligible when the device cluster is as small as that used in the figure.

6. Concluding Remarks

This paper considers an IoT network supporting both the centralized signal transmission and D2D communications. With asymmetric antenna configurations, we propose the
transmit and receive BF for the virtual MIMO formed by D2D cooperation. For the proposed method, we provide the strategies to improve the sum rate of the device pair and a distributive algorithm for device pairing and resource sharing to maximize the network throughput while meeting each device’s desire. It is confirmed that the distributed algorithm works well in the network with transmit CSI also, providing a performance close to that obtained with optimal pairing. The transmit CSI improves the performance of the network significantly even without D2D cooperation but the D2D cooperation leverages the performance further. The performance gain of the D2D cooperation becomes more prominent when the D2D links are relatively better than the downlink by using a larger bandwidth for the D2D links and by applying the method to clusters of closely located devices but farther from the AP.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (nos. NRF-2015R1A2A2A01005390 and NRF-2015R1C1A1A01052529).

References

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," IEEE Internet of Things Journal, vol. 1, no. 1, pp. 22–32, 2014.
[2] J. Lee, K. D. Singh, Y. Hadjadj-Aoul, and N. Kumar, "Wireless and mobile technologies for the internet of things," Mobile Information Systems, vol. 2016, Article ID 8206548, 2 pages, 2016.
[3] M. Hasan, E. Hossain, and D. Niyato, "Random access for machine-to-machine communication in LTE-advanced networks: issues and approaches," IEEE Communications Magazine, vol. 51, no. 6, pp. 86–93, 2013.
[4] X. Lin, J. G. Andrews, A. Ghosh, and R. Ratasuk, "An overview of 3GPP device-to-device proximity services," IEEE Communications Magazine, vol. 52, no. 4, pp. 40–48, 2014.
[5] M. Tehrani, M. Uysal, and H. Yanikomeroglu, "Device-to-device communication in 5G cellular networks: challenges, solutions, and future directions," IEEE Communications Magazine, vol. 52, no. 5, pp. 86–92, 2014.
[6] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," Institute of Electrical and Electronics Engineers. Transactions on Information Theory, vol. 50, no. 12, pp. 3062–3080, 2004.
[7] C. T. Ng, N. Jindal, A. J. Goldsmith, and U. Mitra, "Capacity gain from two-transmitter and two-receiver cooperation," IEEE Transactions on Information Theory, vol. 53, no. 10, pp. 3822–3827, 2007.
[8] J. S. Wang, Y. H. Kim, I. Song, P. C. Cosman, and L. B. Milstein, “Cooperative relaying of superposition coding with simple feedback for layered source transmission,” IEEE Transactions on Communications, vol. 61, no. 11, pp. 4448–4461, 2013.

[9] P. Li and S. Guo, Cooperative Device-to-Device Communication in Cognitive Radio Cellular Networks, Springer, Berlin, Germany, 2014.

[10] J. Jiang, J. S. Thompson, and H. Sun, “A singular-value-based adaptive modulation and cooperation scheme for virtual-MIMO systems,” IEEE Transactions on Vehicular Technology, vol. 60, no. 6, pp. 2495–2504, 2011.

[11] J. Jiang, J. S. Thompson, H. Sun, and P. M. Grant, “Performance assessment of virtual multiple-input multiple-output systems with compress-and-forward cooperation,” IET Communications, vol. 6, no. 11, pp. 1456–1465, 2012.

[12] S. H. Lee, D. R. Shin, H. W. Jeong, and Y. H. Kim, “Distributed bargaining strategy for downlink virtual MIMO with device-to-device communication,” IEEE Transactions on Communications, vol. 64, no. 4, pp. 1503–1516, 2016.

[13] Y. Jeong, J. Yoon, Y. H. Kim, and S. H. Lee, “Distributed matching and resource sharing for space division multiplexing assisted by D2D cooperation,” in Proceedings of the International Conference on Information and Communication Technology Convergence (ICTC ’16), pp. 112–114, Jeju-do, Korea, October 2016.

[14] J. Kleinberg and E. Tardos, “Balanced outcomes in social exchange networks,” in Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing (STOC ’08), pp. 295–304, ACM, New York, NY, USA, 2008.

[15] M. Bayati, D. Shah, and M. Sharma, “Max-product for maximum weight matching: convergence, correctness, and LP duality,” IEEE Transactions on Information Theory, vol. 54, no. 3, pp. 1241–1251, 2008.

[16] Y. Kanoria, M. Bayati, C. Borgs, J. Chayes, and A. Montanari, “Fast convergence of natural bargaining dynamics in exchange networks,” in Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1518–1537, San Francisco, Calif, USA, January 2011.

[17] M. Shamaiah, S. H. Lee, S. Vishwanath, and H. Vikalo, “Distributed algorithms for spectrum access in cognitive radio relay networks,” IEEE Journal on Selected Areas in Communications, vol. 30, no. 10, pp. 1947–1957, 2012.

[18] S. H. Lee, M. Shamaiah, H. Vikalo, and S. Vishwanath, “Message-passing algorithms for coordinated spectrum sensing in cognitive radio networks,” IEEE Communications Letters, vol. 17, no. 4, pp. 812–815, 2013.

[19] D. P. Bertsekas, “Auction algorithms for network flow problems: a tutorial introduction,” Computational Optimization and Applications, vol. 1, no. 1, pp. 7–66, 1992.

[20] T. M. Cover and J. A. Thomas, Elements of Information Theory, John Wiley & Sons, Hoboken, NJ, USA, 2nd edition, 2006.

[21] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt, “Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels,” IEEE Transactions on Signal Processing, vol. 52, no. 2, pp. 461–471, 2004.

[22] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” IEEE Transactions on Information Theory, vol. 47, no. 2, pp. 498–519, 2001.

[23] I. Sohn, S. H. Lee, and J. G. Andrews, “Belief propagation for distributed downlink beamforming in cooperative MIMO cellular networks,” IEEE Transactions on Wireless Communications, vol. 10, no. 12, pp. 4140–4149, 2011.
Research Article
Multitiered and Distributed WSAN for Cooperative Indoors Environment Management

Taewook Heo,1 Hyunhak Kim,1 Yoonmee Doh,1 Kwangsoo Kim,1 Jongjun Park,1 Naesoo Kim,1 JongArm Jun,1 and JeongGil Ko2

1Electronics and Telecommunications Research Institute, Daejeon, Republic of Korea
2Ajou University, Suwon, Republic of Korea

Correspondence should be addressed to JeongGil Ko; jgko@ajou.ac.kr

Received 8 December 2016; Accepted 22 March 2017; Published 4 May 2017

Academic Editor: Shuai Hao

Copyright © 2017 Taewook Heo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

For the past decade, wireless sensor networks have focused primarily on data collection. As a result the network topology for these systems was usually heavily centralized. However, for these networks to form a full system, the introduction of proper actuation units and decision-making intelligence is inevitable. Such a new wireless sensor and actuator network system enables new architectural research issues that have not been previously studied. In this work, we introduce the DWSAN system architecture, which effectively combines both sensor and actuation hardware devices to a single network and manages this network so that the actuation decisions are made in a distributed manner and the topology of the network maintains a multitier architecture. Our intensive set of evaluations reveal that, compared to the centralized approach that has been used in most wireless sensor network systems until now, when actuation units are introduced to the system, the DWSAN architecture reduces the transmission load of the network and the actuation decision-making latency by close to twofold and threefold, respectively. Furthermore, we show that this benefit naturally leads to better scalability of the system, making it suitable for various sensing applications in different environments.

1. Introduction

As the research in wireless sensing systems start to mature with the support from various successful system deployments in a diverse set of application domains, researchers (and now the everyday users of these systems) have started to aim at higher research goals to make the system even more intelligent and suitable to the target applications. A natural next step request for systems that reliably collect sensing information is to autonomously make decisions on how the environment should change, with respect to the collected data [1]. For applications that do not require global knowledge, this is a plausible design choice. Furthermore, these systems should now add actuation units that perform these decisions that are made [2]. Although this may seem like a trivial change to the system, the addition of system intelligence and proper actuation units change the way we think of wireless sensing system development (and deployments) in a new direction [3]. Despite so, most recent work from the wireless sensor network research community still remains to put focus on diversifying its application scope with data collection-based systems (e.g., [4, 5]) rather than detailing the requirements for wireless sensing and actuation system deployment, which, again, is an unavoidable next step.

Specifically, on a technical perspective, the addition of actuation units in a wireless sensing system introduces the need for notifying the target actuators with the proper actions to take. For example, an air conditioning unit may need to change its target temperature level or modify the wind speed, or a lighting unit may be asked to change the illumination level to account for changes in the target environment. These types of actions not only require the packets in a wireless sensing system to travel bidirectionally (e.g., sensor to the gateway and back to an actuator), apart from the unidirectional data collection-centered approach of wireless sensor networks, it also requires further optimizations to maximize the efficiency of message transmissions. Issues arise not only at the protocol level, but also on a topology management perspective. The architecture of how nodes with different capabilities are logically connected can determine
the scalability, or even the usability of the system in different applications. Nevertheless, the answers to such issues are still unknown or have only been proven using simplified preliminary studies [6, 7].

To provide answers to such questions raised by these new types of networked systems, in this work, we introduce a system architecture, where sensor nodes, actuation units, servers, and user devices interconnect (and interact) in a distributed manner to form a multitiered network. Specifically, hardware components in our system, the sensing unit, the actuation unit, the master (gateway) node, and the server, form a hierarchical network architecture, which is designed so that the amount of network traffic and packet delivery latency on the resource limited platforms are minimized. At the same time, this architecture allows for maximum scalability on an entire network's perspective. We generalize such a network architecture as a distributed wireless sensor and actuator network (DWSAN) and present a series of application domains where the DWSAN design can significantly benefit. Furthermore, using an intensive set of evaluations, we show that the distributed and hierarchical network architecture of DWSANs can significantly improve the efficiency of the wireless system.

Specifically, this paper makes the following contributions.

(i) We present a set of hardware implementations for designing a DWSAN, including sensor nodes and gateway nodes.

(ii) We present software techniques to effectively support DWSANs on a contextual basis. Specifically, we propose a new objective function for IETF RPL routing so take sensing similarity as a routing metric.

(iii) The proposed DWSAN protocols and hardware are evaluated through an extensive set of simulations and real implementations.

The remainder of this paper is structured as follows. In Section 2 we present the architecture of the DWSAN system and introduce the benefits of using such a multtiered, distributed networking architecture for various wireless sensing and actuation applications. We introduce the hardware and software components designed and implemented for the DWSAN system in Section 3 and evaluate the performance of our proposal in Section 4. Finally, we position our work along with some related work using Section 5 and conclude the paper in Section 6.

2. Multitiered DWSAN System Architecture

2.1. DWSAN Architecture. While many sensor network architectures exist, the main goals in designing the distributed wireless sensor and actuation network (DWSAN) system architecture can be summarized in threefold. First is to make sure that the network architecture is scalable to large numbers of sensors and actuation devices. Second is to make sure that the response latency of actuation units is minimized as the environmental sensing value changes are detected. Finally, despite the increasing number of data generation units (e.g., sensor and actuation nodes), the DWSAN system should minimize the amount of wireless traffic.

To achieve such goals, we design the DWSAN system as illustrated in Figure 1. Note that, compared to a traditional sensor network, instead of assigning sensors with the goal of reporting all their data to the server directly, the sensing units report their measurements to interested actuators. For
example, in a smart building with automated HVAC and lighting, there is no need for the HVAC actuation unit to know the current illumination level in a target room, nor does the lighting need knowledge of the current room temperature. Furthermore, since real-time temperature data and illumination level data are (in most cases) not necessary at the server, these sensing values are sent to the HVAC actuators and lighting units, respectively, so that they can make their own actuation decisions locally. At the end of the day, as the actuators collect data from multiple sensors that are associated with itself, the actuator nodes send data “summaries” to the server so that the system manager can keep track of the environment.

In many previous systems [8], despite having different roles at heterogeneous nodes, all actuation decisions were made at the server and resent to the network towards a target actuator. Such process saturates the wireless channel quickly due to the server acting as the sink of all the data in the network. By taking a distributed approach, DWSAN tries to distribute the decision-making processes locally and, in turn, spatially distributes the packet loads as well.

2.2. Applications. The following paragraphs introduce applications that can benefit from the use of the DWSAN system architecture.

Room-Level Lighting Applications. Sensors that measure the current illumination level of rooms can be used to imply the comfort level with respect to the desired illumination level of the users. Based on this information, actuators such as the lighting system itself or the blinds on the windows can be interconnected to provide the desired level of light into the room environments. In addition to illumination sensors, other sensors such as passive infrared (PIR) and ultrasound sensors can be used to imply other activities in the room and configure the lighting level with respect to the activity that is occurring in the target environment.

HVAC Applications. By tightly configuring HVAC related actuation units with various wireless sensors, the HVAC system can react in a more fine-grain manner with respect to the real-time conditions of the environment. While a variant of such systems is already implemented in a centralized manner, modifying the architecture in a distributed way can maximize the efficiency of the system.

Other Environment Customization Applications. Besides lighting and HVAC, by using sensors that extract the activity context of a given environment, wireless sensing systems (with intelligence) can be used to estimate the users’ intents. With such base systems, based on the conclusion of such decision-making units, various actuators can be used to control the environmental parameters in many aspects to meet the satisfaction level that the users desire.

2.3. Architectural Benefits. The applications above can be implemented using current-day technologies. Nevertheless, DWSAN can provide a set of technical benefits. Using the following paragraphs we examine some of these benefits, of which we will later evaluate some in Section 4.

Improving the Actuation Response Latency. For many applications, the response time from detecting changes in the environment and performing the required actions to reconfigure the environment can be a critical factor. For example, changes in the illumination level would require quick adaptation of lighting units to assure that the room always stays at the desired illumination level. This is also true for many surveillance applications.

DWSANs configure the network so that nodes form clusters that interconnect related sensor nodes with contextually correlated actuators. Furthermore, sensors directly report their measurements to actuation units rather than a server. This naturally reduces the number of transmission hops and thus reduces the actuation latency. In other words, since actuators make decisions on their actuation in a distributed manner, the data does not need to be forwarded to the central server to reduce the time from the point of sensing the environmental changes to the time of actuation, allowing systems to react more rapidly.

Reducing the Wireless Traffic on the Shared/Limited Wireless Medium. Most of the widely used wireless bands are shared by different applications, and even different wireless standards. To allow other systems to operate effectively, it is important to reduce the traffic overhead of the DWSAN system. The DWSAN system due to the reduced number of hops (or transmissions) it takes to deliver the sensing data to the final destination naturally reduces the amount of traffic that is put in the wireless medium. This allows room for other systems and protocols to interoperate well in the same geographical environment.

Maximizing the System’s Scalability. Most environmental sensing applications can benefit by the use of more sensors deployed densely in a target environment. However, this is not easy to achieve due to the limited wireless bandwidth. By decreasing the number of transmitted packets, the system can use the remaining wireless capacity to support more nodes. Since centralized data gathering causes communication bottleneck, DWSAN distributes the role of decision-making to different devices to further increase the system’s scalability.

3. System Design

We now present the hardware and software components that constitute the DWSAN system architecture.

3.1. Hardware Components. In terms of hardware, DWSAN consists of four major components: (1) sensor module, (2) actuation module, (3) master gateway module, and the (4) back-end server component. The following paragraphs will present the components in the hardware in detail and discuss about the design choices for each major hardware component. We note that all our wireless components in these components use the Atmel RF212 radio on the 900 MHz band [9]. We select to use the sub-GHz range for our wireless communications since it provides us with a more stable communication range compared to the 2.4 GHz ISM band,
which is important, given the dynamics of various indoor environments.

**Sensing Platform.** For hardware designs, our primary focus was to design a generic sensing platform that could be applied to various indoor applications. Therefore, we designed a custom sensing platform as presented in Figure 2. Specifically, as sensing components, this platform includes a PIR sensor, CO and CO$_2$ sensors, an illumination sensor, and a set of temperature and humidity sensors. Based on our literature survey of indoor sensing applications, we decided that these sensors were the ideal set. Nevertheless, by exposing a set of additional GPIOs on the board, we allow the platform to be expandable towards other sensing modalities as well. The current version of this sensing platform used the Atmel 1284p microcontroller and provides buttons, LEDs, and USB connections.

**Actuation Platform.** Using the information collected from the sensing units, the role of the actuation platform is to control electronic (actuation) devices effectively. For example, if sensors in the environment report that the temperature of the room is higher than the target temperature, related actuation units (e.g., air conditioner or fan) try to configure themselves so that the target temperature is maintained. In reality, a diverse set of actuator types will exist, one (or more) for each application. One example of such application driven actuation unit would be the light control unit as Figure 3(a) shows. This unit is connected to the wall-mounted light controller to wirelessly manage lighting in a room scale.

Apart from such application-targeted devices, we also designed a more generic platform that controls the power state of the connected device (cf. Figures 3(b) and 3(c)). Using this electricity controlling unit the system can manage the power states of various appliances.

**Master (Gateway) Node and Back-End Server.** The gateway or master node of DWSAN, as presented in Figure 4, acts as the root of the IETF RPL network that we use to manage the nodes as a single network. This node connects the network of sensors and actuators to the server node. However, due to the distributed nature of DWSAN, the master node’s main role is network management rather than data forwarding. The master node includes its own Atmel RF212 radio to connect with the wireless network and this module is attached to an Internet connected OpenWRT box to establish and maintain UDP/TCP connections with the back-end server, which is a simple PC for installing a lean-server design.

**Back-End Server.** The selection of the hardware components for the server was affected significantly by the DWSAN system architecture. Specifically, the distributed processing nature of DWSAN allows the system to utilize a lean-server: allowing us to install a simple PC scale server machine. While for many applications the processing overhead may not be a fundamental issue, with scaled-up systems, we argue that our DWSAN architecture reduces the potential of server-side overflow of processing. A detailed illustration of the server architecture is presented in Figure 5.

3.2. Software Components. On the software perspective, the implementation of DWSAN includes a set of interesting features in which some are essential to maintain the distributed, multilayered architecture and some are implemented to further maximize the benefits that DWSAN introduces. The following paragraphs discuss about these functionalities in detail.

**Profile Generation and Distribution.** In designing a distributed decision-making system, one of the most important factors to consider is the effective assignment of roles for all devices in the system. In fact, this role distribution and assignment process is the core of breathing in the necessary intelligence to a distributed system. In DWSAN, we allow system operators to input target sensing granularities and actuations using simple grammars and deliver this information to all nodes in the network using sensing/actuation profile messages. We follow the specifications of the CoAP application layer protocol [10] in designing the format of the profiles. Once these messages are formatted at the server, they are distributed via *unicast* to each node in the network. For each unicast, we request for a software level, end-to-end acknowledgment message to ensure that the packet was successfully sent. While we agree that this process of unicasting profiles to each node may be a burdensome process, we take such design choice due to the fact that the delivery of this profile message is, by far, one of the most important processes, since it determines how the sensor nodes should sample their sensors and how actuators should behave with respect to incoming sensing data. Despite the distribution itself being a time-consuming
process, this happens only once when the system is initially installed. For additional nodes that are installed and updated during runtime, the server selectively sends profile messages to related nodes only; thus, the time and wireless channel capacity overhead can be reduced to be minimal.

**RPL Compatible Network with Clusters.** In the proposed DWSAN system we utilize a standardized routing protocol for low-power and lossy networks (LLNs). Specifically, we find the IETF IPv6 routing protocol for LLNs, RPL [11], suitable for our network configurations due to three main reasons. First, using the RPL routing protocol forces the implementations to be standard-compliant at the networking and link adaptation layers. Second, by utilizing IPv6 addresses, DWSAN-based systems can easily connect with the larger Internet infrastructure and allow controlled access from external IPv6 machines. Lastly, RPL is designed to support bidirectional traffic patterns; thus, it allows point-to-point traffic to take place effectively and eases the development of a closed feedback-loop system.

While utilizing RPL the way it is can support most of the functionalities the DWSANs require, the performance (or the network efficiency) may suffer. Sensor nodes that are associated with a given actuator node may not have direct routing path connectivity. In other words, while having physical single hop coverage, due to the RPL objective function (OF) (objective functions (OFs) in RPL determine how a next hop node in the multihop routing path should be selected; more details on this topic can be found in previous work by Ko et al. [12]), a packet may need to travel more hops to reach its (logically) associated actuation node. To resolve this inefficiency, we introduce a new RPL OF. Specifically, the Context Clustering OF defines a set of contextual targets and clusters nodes with similar (or related) contextual measurements. We do this by allowing nodes to overhear messages transmitted by potential parent nodes (via overhearing) and compare the trend of data changes using a simple cross-correlationalgorithm. With correlations >0.7 we declare a high similarity on the two data/sensing streams. By specifying the supported sensor types on a sensor and the desired sensor measurement types at the actuator, next hop nodes in RPL are selected so that sensor nodes and actuator nodes with contextual similarity are clustered together. If possible (in the case where the node is within communication range), this actuator is selected as the next hop node for the sensors. Here, we note that sensors may select multiple actuators, as the same sensor data may be required at different actuators. In this case, sensing values can be sent either via unicast to each associated actuator sequentially or via multicast to all associated actuators.
simultaneously. Overall, this new objective function observes the contextual similarity of data being collected at the sensors to determine the next hop node so that measurements can be suppressed as the packet traverses the network. By utilizing the Context Clustering objective function, we point out that the packet traverse path length from geographically relevant sensor nodes to target actuator nodes can be minimized. This naturally reduces the number of packet transmissions in a network and thus increases the scalability of the system.

**Actuator Collaborations.** There are situations in which multiple actuators that form a single actuation system (or unit) need to collaborate together to perform the proper actuation required at the environment level. For example, multiple lighting units may need to collaborate to provide the proper illumination required in a room. For such cases, DWSAN allows the actuation units to form a subnetwork and select an internal master node. To ease the decision-making process in such scenarios, all related sensing data is forwarded to the head actuator in the subnetwork (which is predefined with respect to the application's target) and this head node sends actuation commands to control other actuators on how they should act.

This subnetwork of actuators is configured in two different ways. First, the application administrator can specify groups using the profile messages discussed earlier. Or second, the nodes can compute the contextual similarity of the data they receive to autonomously compute related actuation nodes. In the current implementation of DWSAN, we select the first approach of explicitly selecting the master node using profile messages but plan to address the issue of autonomous group management as part of our future work. Specifically, potentially we can allow nodes to compute similarities and identify the most common parent node that shares the highest contextual similarity for this automation process.

**Adaptive Sampling for Indoors Environments.** On a data analysis perspective, gathering fine-grain data from a sensor network is always the best scenario. However, in sensor networks, nodes only hold a limited amount of resources in terms of energy, and bandwidth. Under such circumstances, sending all the data to the gateway is not a practical approach. With this in mind, many efforts have been made to reduce the number of transmitted sensing samples while still representing the original data properly.

A popular way to approach this issue is using compressive sensing [13, 14], in which nodes gather information from other nodes in the same region to configure their sampling and transmission rates (e.g., using spatial correlation). While this is attractive, we point out that, in indoor environments, distributed sensors can show significantly different measurements with respect to their relative locations to the actuators (e.g., air conditioner, lighting sources), making spatial correlation difficult. Furthermore, since practical systems will not deploy tens of sensors in a single room, the correlation process becomes even more challenging. To exemplify, we plot in Figure 6 temperature readings from Sensirion SHT11 sensors at two locations in an office, one positioned close to the air conditioning unit and the other 10 meters away. Notice that while the two sensors show somewhat similar behaviors, differences in location and surrounding environment factors cause the plots to show differences in the amplitude variation and short term trends for some regions. This gives us a first-hand impression that using spatial correlation can be challenging in indoor environments and suggests the need for a data summarizing scheme that operates only based on the sensors’ local data.
Besides using spatial correlation, there have also been attempts to summarize the original sensor stream using the data's temporal characteristics. These attempts, studied by many researchers in diverse domains, include adaptive/probabilistic models for data prediction which require a sufficient amount of computing resources at the nodes [15] and event/threshold-based data summarizing schemes [16]. In DWSANs we incorporate a similar approach to the latter. In our scheme, once a node collects a measurement from its sensor, it first passes this measurement through a low pass filter. While this low pass filter may smooth some of the sensor dynamics, we apply this filter so that small variations in the filter. While this low pass filter may smoothen some of the sensor, it first passes this measurement through a low pass filter. We configured a 100-node network in a 100 × 100 meter grid (communication range set to 20 meters) while keeping 10% of the nodes as actuators with the rest being sensor nodes. Nodes’ locations were set to be random and the transmission delay (actual time to deliver a packet over the wireless network) required to make an actuation decision at the actuator after a sensing sample being sent, (2) we measure the amount of wireless traffic generated in the network with various data reporting configurations, and finally, (3) we measure the scalability of the system in terms of packet reception ratio as the number of supported nodes grows in the network.

4. Evaluation

In this section we present performance benefits that the DWSAN architecture brings along with the wireless traffic reductions that our proposed adaptive sampling scheme introduces.

We start by presenting our evaluations for the DWSAN system using both Matlab simulations and data collected from real hardware-based experiments. While we agree that simulations only partially capture the effect of real wireless conditions, we utilize simulations for experiments that require large-scale deployments. Matlab simulations were done in Simulink by mimicking similar network configurations as our target deployments later discussed.

For our evaluations we focus on three different performance metrics. Specifically, (1) we analyze the latency required to make an actuation decision at the actuator after a sensing sample being sent, (2) we measure the amount of wireless traffic generated in the network with various data reporting configurations, and finally, (3) we measure the scalability of the system in terms of packet reception ratio as the number of supported nodes grows in the network.

4.1. Sensing-to-Actuation Response Latency. As mentioned in Section 2.3, the actuator’s response latency to sensing measurements can be a critical factor for many applications. Using simulations, we show in Figure 7 the latency performance of the DWSAN network compared to a centralized networking approach.

We configured a 100-node network in a 100 × 100 meter grid (communication range set to 20 meters) while keeping 10% of the nodes as actuators with the rest being sensor nodes. Nodes’ locations were set to be random and the transmission delay (actual time to deliver a packet over the wireless system using both Matlab simulations and data collected from real hardware-based experiments. While we agree that simulations only partially capture the effect of real wireless conditions, we utilize simulations for experiments that require large-scale deployments. Matlab simulations were done in Simulink by mimicking similar network configurations as our target deployments later discussed.

For our evaluations we focus on three different performance metrics. Specifically, (1) we analyze the latency required to make an actuation decision at the actuator after a sensing sample being sent, (2) we measure the amount of wireless traffic generated in the network with various data reporting configurations, and finally, (3) we measure the scalability of the system in terms of packet reception ratio as the number of supported nodes grows in the network.

4.1. Sensing-to-Actuation Response Latency. As mentioned in Section 2.3, the actuator’s response latency to sensing measurements can be a critical factor for many applications. Using simulations, we show in Figure 7 the latency performance of the DWSAN network compared to a centralized networking approach.

We configured a 100-node network in a 100 × 100 meter grid (communication range set to 20 meters) while keeping 10% of the nodes as actuators with the rest being sensor nodes. Nodes’ locations were set to be random and the transmission delay (actual time to deliver a packet over the wireless

Figure 6: Empirical temperature measurements collected from two different locations in an indoor environment.

Figure 7: Sensing-to-actuator decision and response latency for DWSAN and a centrally managed network.
medium) was set to $\sim 3$ ms (assuming a 100-byte packet on a 250 kbps link). Furthermore, since different applications require different per-hop packet processing, we varied the processing latency of each packet by 10, 20, and 40 ms. The gateway was set to be at the center of the random topology, which is the best-case scenario for the centralized scheme. On an operational perspective, sensor nodes send their measurements to the decision-making device (in DWSAN the associated actuator and in the centralized scheme the gateway) to measure the latency required in making actuation decisions. For fair comparisons, we neglect the random channel loss. Nevertheless, even when considered, it would not affect the relative performance of the two schemes.

Notice in Figure 7, where we plot the mean and standard deviation over 100 different random topologies, that DWSAN outperforms the centralized scheme by more than twofold. This trend increases further with increasing per-node processing latency. This suggests that DWSAN successfully delivers its packets to actuators effectively and the fact that they traverse over smaller number of hops reduces the decision-making latency. Note that DWSAN may encounter an increased level of overhead on a per-hop basis. Nevertheless, if we compare the 10 msec case for centralized and 20 msec case for DWSAN, DWSAN still shows better performance.

### 4.2. Analyzing the Required Wireless Packet Traffic

Another important aspect is the usage of the wireless medium. With less packets in the air, the network will be able to scale to more nodes and gain a more fine-grained view of the target environment. For this purpose, using the same environment as above, in Figure 8, we plot the average number of sensor samples (packets) sent in the network computed over 100 tests for a 60-minute test (simulated) duration each. In this experiment however, we try varying the number of actuators in the network. Specifically, we test for cases where 10, 15, 20, and 25% of the nodes in the network are actuators and the others are sensors. We note that sensors send messages at an interval of 1 packet per second (pps) while actuators report their actuation summaries to the server at 0.1 pps.

In Figure 8(a) we keep the location of the gateway at the center of the topology as before. Again, this is the most ideal case for the centralized scheme since it minimizes the path length for the packets. Even so, notice that DWSAN outperforms to the centralized scheme slightly. This trend continues even when the number of actuators in the network increases. This is surprising given that the increased number of actuators should further minimize the path length and thus achieve a significantly lower packet transmission count for DWSAN. To explain this, we note the fact that the single hop communication radius was set to 20 meters while the nodes were deployed in a 100-meter × 100-meter area. Therefore, the field was relatively saturated and an increasing number of actuators would only give minimal impact in reducing the number of packet transmissions.

In Figure 8(b) we now vary the locations of the master (gateway) node randomly. As expected, compared to Figure 8(a), the number of packet transmissions for the centralized scheme increases by $\sim 40\%$, whereas the performance of DWSAN is not significantly impacted. Overall these results suggest that the DWSAN architecture can successfully minimize packet transmissions (e.g., channel saturation) effectively.

### 4.3. Impact of Adaptive Sampling

We now expand this topic and introduce the impact of the adaptive sampling scheme (cf. Section 3.2) in reducing packet transmissions. We note that a preliminary version of the scheme below has been presented as part of reference [18]. For our evaluations, we perform a per-second measurement of temperature and humidity indoors for a week using the hardware introduced in Section 3.1.

We plot in Figure 9 the percentage of samples points required to reconstruct the original per-second-scale measurement when applying the adaptive sampling scheme with and without the trickle timer reports for different $\Delta_{TH}$. Quantitatively speaking, the original sample set consisted of $\sim 520$ K samples for each sensor type. Using the simple adaptive sampling scheme we were able to represent the entire data stream using as low as 0.019% of the original data (e.g., 100 samples) for the temperature data when $\Delta_{TH} = 1.0 ^{\circ} C$. We can also see a similar trend for humidity.

Given the high compression ratio, the next question is how accurate this compressed measurement is compared to the original high-rate data. For this Figure 10 compares the mean error of our scheme’s summarized reports with a periodic scheme that uses the same number
of reports but evenly distributed across the entire data set (Periodic: Adaptive). In other words, this second method represents a case where the samples were made at a periodic interval and the spacing between the samples was computed with respect to the number of samples from the adaptive scheme. For both schemes, at a given time, the system takes the most recent sample as the current temperature or humidity value. We compare the contextual values of both schemes against the actual fine-grained data collected at 1 sample per second. By comparing Periodic: Adaptive against Adaptive and Periodic: Adaptive + Timer against Adaptive + Timer in Figure 10, we can notice that the nonperiodic schemes (e.g., our propose schemes) show lower errors when compared to the original data set. This is mainly due to our proposed scheme effectively detecting regions where the measurements dynamically change.

Overall, as Figure 11 shows, our scheme effectively tracks and reports measurements (mostly) during the dynamic regions, thus, significantly increasing its accuracy while reducing the amount of traffic on the wireless medium.

4.4. Network Scalability. Lastly, we examine the scalability of the system in terms of the packet reception ratio (PRR) observed at the gateway with increasing number of nodes in the network. In the same geographical setting as the other simulations, we increase the number of nodes in the network to as high as 400, while configuring 10% of these nodes as actuator nodes. We plot the mean PRR and the standard deviation computed over 100 independent runs in Figure 12.

We take $\Delta_{TH} = 0.4$ and set the minimum trickle timer value to 500 msec with a maximum reporting interval of 60 mins. Notice in Figure 12(a) that, with a data rate of 1 pps, both the DWSAN and the centralized approach fail to provide quality service with >200 nodes. Nevertheless, by effectively reducing the packet transmissions, DWSAN achieves higher PRR than the centralized approach. The difference in PRR becomes more prominent in Figure 12(b) as we use a lower data rate of 0.4 pps at each node. In this case, we can see that DWSAN provides 90+% PRR for all cases and the performance degrades gracefully while the PRR of the centralized scheme degrades severely as the number of nodes exceeds 100.

To summarize our evaluations, we can easily notice that the distributed and multitiered nature of the DWSAN system architecture is effective in reducing the hop count towards the decision-making device and therefore reduces the sensing-to-actuation latency and minimizes the number of packets being transmitted in the wireless medium and this naturally leads to supporting node wireless nodes in the same geographical area compared to the centralized approach.

4.5. Real-World Deployments. The DWSAN system was deployed as part of two pilot studies. The first study, as illustrated in Figure 13, was made at our research institute. The purpose of this study was to study the effectiveness of our proposed distributed and multitiered network, while at the same time, we put focus on evaluating the application-side of the system as well. Specifically, we selected autonomous lighting as the application for our system, where heterogeneous sensors, such as passive infrared (PIR), microphone, and illumination sensors were used to determine the activity states in a meeting room environment [19]. This deployment consisted of eight sensing units connected to six actuation (LED lighting) units. Despite the small system configuration, where all nodes were physically connected via single hop, we were able to notice that the proposed DWSAN system architecture reduced the number of packets transmitted in the air by ~50% compared to a IETF-RPL-based single-tier network architecture [12]. Deeper investigation on the traces revealed that ~10% of this 50% was from the network architectural changes while the adaptive sampling contributed to the other 40%.

Based on the preliminary experiences from the internal testbed, we then performed a second pilot study in a 13-story commercial building in the Gangnam District of Seoul, South Korea. Our deployments, again, as Figure 14 shows, were designed for a smart lighting application, covered two floors, and included 20 sensing units (e.g., a mixture of PIR, microphone, illumination, and power-usage monitoring sensors) with 14 LED dimmer actuation units. Compared to the internal testbed in Figure 13, this deployment was for an environment with more frequent human activity and not all nodes were reliably connected via single hop connections. As a result, we were able to notice the impact of the multitiered and distributed decision-making process of DWSANs to be even more effective. Specifically, we observed ~65% reduction of network traffic by the use of the DWSAN architecture in this environment.

5. Related Work

Finally, we introduce a few previous works that deal with the efficient design of wireless sensor and actuator networks (WSANs). We introduce the contributions of these previous works and also discuss about their differences with the contributions made in this work with the DWSAN architecture.

In [7], Li discusses the concept of WSANs and prototype of a light sensing and actuating application for a home environment. This paper also introduces the benefits
of maintaining a layered networking approach, which is something similar to the multitiered approach introduced in this work. Nevertheless, the work by Li, being one of the first steps in introducing actuation to sensing systems, fails to evaluate the performance of the system in a large scale. On the other hand, we set up an extensive set of simulations to show the effectiveness of maintaining a distributed and multitiered WSAN.

Cañete et al. apply a service oriented approach to WSAN system designs [20]. This introduces interesting framework and middleware architectures with unique and useful characteristics. Nevertheless, compared to our work, the authors focus mostly on introducing the design of the system framework while we perform experiments that showcase the benefits of using DWSANs in a quantitative way.

In the work by Wu and Rowe [21], the authors introduce SAN-Logic, a lightweight logic-based programming paradigm, to enable the dynamic programmability and configuration of WSANs. The authors show that the use of this logic-based WSAN design can result in optimized network architectures that reduce the packet transmissions in the network. While this is an attractive approach and can be applied to various scenarios, compared to our proposed DWSAN, the resulting network is not standard-compliant nor can it easily and effectively account for changes in the topology or network environment characteristics.

Lastly, most similar to our work is the work presented by Boukerche et al. [6]. This work introduces the concept of centralized and decentralized networks for WSANs and discuss about the benefits of the two different approaches. To construct the decentralized network, the authors introduce the Periodic, Event-driven, and Query (PEQ) based protocol, which essentially ties geographically closely related sensor nodes to the nearest actuator node. The downside of this approach is, however, that the authors fail in discussing how sensors with different contextual information would interact. In our RPL-based approach discussed in Section 3.2, we account for heterogeneous sensor types by making the format of the objective function flexible enough, not to mention the standard-compliance with other RPL networks. Since the increase in sensing modalities and the interaction with various standard-compliant systems is unavoidable as WSANs gain more and more interest, the DWSAN architecture introduces a more advanced and expandable system architecture.

6. Summary

In this work we introduce the DWSAN system architecture designed for various indoors sensing and actuation-requiring applications. The DWSAN architecture exploits its distributed and multitiered characteristics to form a networked system that effectively reduces the packet transmission count and sensing-to-actuation latency, while maximizing the scalability of the network. We introduce software and hardware components that we designed for the DWSAN system and show using both simulations and real data traces from our custom hardware that the performance of DWSAN outperforms that of a centralized networking approach, which is still widely used in various monitoring and actuation applications. We believe that this study is an effort to finally diversify our thoughts on wireless sensing system architectures and quantitatively notice that, with new capabilities added to the wireless system, a new perspective of network architecture design is required.
Figure 12: Packet reception ratio observed at the gateway (master) node with increasing number of nodes in the network.

(a) Data rate = 1pps

(b) Data rate = 0.4pps

Figure 13: Internal deployment at research-scale.

Deployment testing of autonomous changing in-room actuation behaviors
(Normal versus presentation mode lighting)

Figure 14: Larger-scale DWSAN deployment in an actively used commercial building.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported in part by Global Research Laboratory Program (2013K1A1A2A02078326) through NRF, DGIST Research and Development Program (CPS Global Center) funded by the Ministry of Science, ICT & Future Planning, and in part by the Korean Ministry of Knowledge and Economy Project no. 10035570 “Development of Self-Powered Smart Sensor Node Platform for Smart and Green Buildings.”

References

[1] I. Akyildiz and I. Kasimoglu, “Wireless sensor and actor networks: research challenges,” Ad Hoc Networks, vol. 2, no. 4, pp. 351–367, 2004.
[2] J. A. Stankovic, “When sensor and actuator networks cover the world,” *ETRI Journal*, vol. 30, no. 5, pp. 627–633, 2008.

[3] S. G. Hong, N. S. Kim, and W. W. Kim, “Reduction of false alarm signals for pir sensor in realistic outdoor surveillance,” *ETRI Journal*, vol. 35, no. 1, pp. 80–88, 2013.

[4] J. Ko, J. H. Lim, Y. Chen et al., “MEDiSN: medical emergency detection in sensor networks,” *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 10, no. 1, article 11, 2010.

[5] D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton, “CodeBlue: an ad hoc sensor network infrastructure for emergency medical care,” in *Proceedings of the MobiSys 2004 Workshop on Applications of Mobile Embedded Systems (WAMES ’04)*, June 2004.

[6] A. Boukerche, R. B. Araujo, F. H. S. Silva, and L. Villas, “Wireless sensor and actor networks context interpretation for the emergency preparedness class of applications,” *Computer Communications*, vol. 30, no. 13, pp. 2593–2602, 2007.

[7] S.-F. Li, “Wireless sensor actuator network for light monitoring and control application,” in *Proceedings of the 3rd IEEE Consumer Communications and Networking Conference (CCNC ’06)*, vol. 2, pp. 974–978, January 2006.

[8] O. Gnawali, K.-Y. Jang, J. Paek et al., “The tenet architecture for tiered sensor networks,” in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys ’06)*, pp. 153–166, ACM, Boulder, Colo, USA, November 2006.

[9] Atmel., “Low Power 700/800/900 MHz Transceiver for IEEE 802.15.4-2006, IEEE 802.15.4c-2009, Zigbee, 6LoWPAN, and ISM Applications,” 2010, http://www.atmel.com/Images/doc8168.pdf.

[10] Z. Shelby, K. Hartke, C. Bormann, and B. Frank, *Constrained Application Protocol (CoAP). Internet Draft*, IETF (work in progress), 2012.

[11] T. Winter, P. Thubert, A. Brandt et al., “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” RFC 6550, 2012.

[12] J. G. Ko, A. Terzis, S. Dawson-Haggerty, D. Culler, J. Hui, and P. Levis, “Connecting low-power and lossy networks to the internet,” *IEEE Communications Magazine*, vol. 49, no. 4, pp. 96–101, 2011.

[13] K. Matthias, L. Thiele, and J. Beutel, “Reconstruction of the correct temporal order of sensor network data,” in *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN ’11)*, April 2011.

[14] X. Wu and M. Liu, “In-situ soil moisture sensing: Measurement scheduling and estimation using compressive sensing,” in *Proceedings of the 11th ACM/IEEE Conference on Information Processing in Sensing Networks (IPSN ’12)*, pp. 1–11, Beijing, China, April 2012.

[15] D. Chu, A. Deshpande, J. M. Hellerstein, and W. Hong, “Approximate data collection in sensor networks using probabilistic models,” in *Proceedings of the 22nd International Conference on Data Engineering (ICDE ’06)*, p. 48, April 2006.

[16] C. Liu, K. Wu, and J. Pei, “An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 7, pp. 1010–1023, 2007.

[17] L. Philip, N. Patel, D. Culler, and S. Shenker, “Trickle: a self-regulating algorithm for code propagation and maintenance in wireless sensor networks,” in *Proceedings of the 1st Symposium on Networked Systems Design and Implementation (NSDI ’04)*, San Francisco, Calif, USA, March 2004.
Research Article

Collecting Big Data from Automotive ECUs beyond the CAN Bandwidth for Fault Visualization

Jeong-Woo Lee, Ki-Yong Choi, and Jung-Won Lee

Department of Electrical and Computer Engineering, Ajou University, Suwon, Republic of Korea

Correspondence should be addressed to Jung-Won Lee; jungwony@ajou.ac.kr

Received 9 December 2016; Accepted 7 February 2017; Published 27 February 2017

Academic Editor: Jeongyeup Paek

Copyright © 2017 Jeong-Woo Lee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A hardware-in-the-loop (HiL) test is performed to verify the software functions mounted on automotive electronic control units (ECUs). However, the characteristics of HiL test limit the usage of common debugging techniques. Meanwhile, the logs of how the program uses memory can be utilized as debugging information collected by the controller area network (CAN). However, when the 32 KB memory is observed with 10 ms period, about 96% of the data on each cycle is lost, since the CAN only can transfer 1.25 KB of data at each cycle. Therefore, to overcome the above limitations, in this study, the memory is divided into multiple regions to transmit generated data via CAN. Next, the simulation is repeated for each divided region to obtain the different areas in each simulation. The collected data can be visualized as update information in each cycle and the cumulative number of updates. Through the proposed method, the ECU memory information during the HiL test was successfully collected using the CAN; the transmission is completed without any loss of data. In addition, the data was visualized in images containing the update information of the memory. These images contribute to shortening the debugging time for developers and testers.

1. Introduction

The software installed in an automotive electronic control unit (ECU) has increased significantly in size and complexity. This automotive software is responsible for features directly connected to the user’s safety, such as the advanced driver assistance system (ADAS) or the adaptive cruise control (ACC). Automotive software is developed using the V-model development process, with corresponding verifying methods in each stage. The verification method is separated into model, software, processor, and hardware-in-the-loop (MiL, SiL, PiL, and HiL) tests, depending on the integration level of each stage. Among them, the HiL test is performed to check whether the functions meet the requirements through the interaction between the software mounted on the completely developed ECU and the peripheral hardware [1].

In the HiL test, the HiL simulator provides input to the target hardware, which is treated as a black box, and observes the corresponding results [2]. When the faults are detected in the test results, the developers should proceed to the debugging. Mostly, debugging the hardware is proceeded by using the interface such as joint test action group (JTAG) to monitor the internal behavior of the target. But, there are some limitations to obtaining the debugging information in the HiL test environment. First, when performing the test with the completed ECU form, it is hard to modify the hardware when debug interfaces, such as JTAG and the background debug mode (BDM), do not exist or are not exposed. Further, stopping the system arbitrarily, for example, to obtain debugging information, is limited. To interrupt the HiL test in progress at any time, both the ECU and the HiL simulator must be stopped at the same time. However, the HiL simulator is configured independently from the ECU; thus, there is a limit to simultaneously operating it at the system-clock level [3].

Meanwhile, the memory information collected periodically during the program execution can be utilized as debugging information. Embedded systems, such as the ECU, execute the actions defined in a program by loading data from memory and storing the updated data. These operations are performed using memory-interface instructions defined in the program [4, 5]. In other words, observing the logs that use the memory can be useful for understanding the program operation. However, since the information generated
regularly in the memory is large, techniques for transmitting the data are required. For example, over 3 MB of data is generated per second when 32 KB of data is generated every 10 ms. To provide the data generated during the HiL test to the developer, the following challenges should be considered.

First, methods are needed for overcoming the network bandwidth limitation. The FlexRay, controller area network (CAN), and local interconnect network (LIN) can be utilized to transfer the large amount of memory data generated inside the ECU while executing the HiL test to the outside [6]. Among them, the FlexRay, which is not commonly used in the ECU, is unsuitable for general use. The LIN cannot be used for large data transmissions because of its low bandwidth: about 20 Kbps [7]. Therefore, the CAN protocol is appropriate for collecting massive memory data [8]. However, since the CAN’s maximum transfer rate is 1 Mbps, the amount of data that can be transmitted in the above environment is 1.25 KB per 10 ms; that is, 96% of the data is lost. An additional technique is required to overcome this limitation.

Attempts have been made to overcome the limitations when transferring large amounts of data using CAN by modifying the CAN protocol or compressing and fragmenting data. However, existing methods are improper for the large data transmissions required in the HiL test environment. First of all, the technique for modifying the CAN protocol requires a revision in the communication driver to identify the modified protocol [9–11]. Secondly, the technique to compress the data can interrupt the existing operation because it inflicts an extra load on the processor. Moreover, when the compression ratio does not satisfy the requirement, the amount of uncompressed data can be lost or communication can be delayed [12–16]. Lastly, in the technique to fragment the data, since the ECU has a limited buffer size, the previously generated, unsaved data can be lost [17].

Meanwhile, by using the transfer technique in Figure 1(a), a large amount of memory data can be transmitted outside of the ECU using CAN by repeating the simulation. Due to the automotive software characteristics that take charge of the user’s safety and require strict real-time operation, the variables for a specific function always operate deterministically [18]. Using this characteristic, the data generated in each separate region is sent by repeating the simulation. The size of each memory region is determined by considering the memory size, the communication environment, and the ECU operation period. The transferred data is stored in the database along with its order information. Using this technique, large amounts of data can be sent outside without requiring hardware modifications or interfering with existing operations.

When providing the memory data collected from the ECU to the developer and tester, it is more effective to provide a visualized image than text [19, 20]. When the collected data is provided in text, they must check the values collected from the 32-KB memory generated every 10 ms. However, by providing an image, the developer can observe the updates of each memory address on a 256 × 128-entry image. Based on the information in the image, the developer can intuitively select a focus area. The collected memory data can be visualized as shown in Figure 1(b). The data collected periodically during the simulation, from $t_0$ to $t_n$, is stored in a database. The memory-update information in $t_1$ can be acquired by comparing it with the data generated in $t_0$. The memory-update information in $t_2$ to $t_n$ can be similarly acquired. By visualizing the acquired memory-update information, it is possible to monitor the operations in the memory via images during the program execution. In addition, it is possible to shorten the debugging time by eliminating unnecessary information that is not related to the operations.

Therefore, in this paper, we propose techniques for collecting the large amount of memory data generated periodically in the ECU during the HiL test and visualizing the text data effectively. First, we segment the memory area into a size that can be transferred per unit time, considering the size of the memory and the communication environment. Then, we repeat the simulation for each segmented area to collect the data generated there. Lastly, we acquire the memory-update
information from the collected data to visualize the update information in each unit time and the cumulative update frequency in each address. By using our proposed method, massive data generated from the ECU can be transmitted to the external database. The collected data can be useful information for the developers to understand the defective behavior of the target.

The rest of this paper is organized as follows. Section 2 explains the limitations to acquiring debugging information in the general HiL test environment and existing methods to overcome the CAN bandwidth. In Section 3, methods to collect memory data using the data-cascading method and the visualization of the collected data are explained. In Section 4, modules to process the proposed methods are described. Section 5 presents the implementation of the methods and the verification of the collected data. The conclusion is presented in Section 6.

2. Related Work

In this chapter, the automotive HiL test, the limitations that occur when acquiring debug information in an environment with restricted hardware modifications, and limitations in utilizing system resources are described. Then, to overcome the transmission limit of the CAN protocol, a conventional method for transferring a large volume of data and its limitations for applying them in the HiL test environment are explained.

2.1. Hardware-in-the-Loop (HiL) Testing. Automotive software is developed in accordance with the V-model development process and has a variety of verification methods, depending on the level of system integration at each stage. Among them, the HiL test takes charge of testing the interaction between the software mounted on the fully developed ECU and the peripheral hardware. The reasons for performing the HiL test are as follows. First, by building a simulation environment, it is possible to reduce the costs incurred in initially establishing and then maintaining a real-car environment [21]. Furthermore, improving the test coverage, increasing the test reproducibility, and shortening the test time can be accomplished through test automation [22].

In general, the HiL test environment consists of a host PC, the HiL simulator, and the ECU. The host PC transfers the test scripts to the HiL simulator; they include use scenarios for running the completely developed software in an automobile according to the testing purpose. The HiL simulator receives the test script from the host PC via a high-speed network such as Ethernet and delivers the test inputs stated in the script to the ECU via I/O or communication interfaces. Lastly, the system under test is composed of the ECU and the peripheral hardware. The ECU in the system receives the inputs based on the test script, performs the operation according to the inputs, and delivers the output to the HiL simulator to determine the presence of faults.

However, obtaining debug information on the occurred faults has the following limitations. Firstly, it is impossible to use commonly used hardware debuggers, such as JTAG or BDM, when the interfaces are either absent or not exposed to the outside [22]. As the testing process proceeds to subject the completely developed ECU, modifications for appending a debugging interface are prohibited. When debugging ports are available to use, the test cannot be stopped arbitrarily to collect data because of the test script which includes the use scenario. To suspend the HiL test, not only the ECU, but also the HiL simulator must be stopped at the exact same time. However, the HiL simulator, which differs from an In-Circuit Emulator (ICE) or a software debugger, is configured independently of the ECU; thus, it is limited to simultaneously controlling both systems at the system-clock level. In addition, utilizing the source code to detect the behavior causing the fault is restricted, because the testing target is treated as a black box.

Meanwhile, embedded systems such as ECUs use memory-interface commands, such as MOVE, LOAD, or STORE, to acquire data stored in the memory for processing the operation and to store the data into memory, for example, RAM. In other words, since RAM is utilized to store the data changed during the operation stated in the test, periodically monitoring the data updates in memory can be used to trace the behavior of the program processed inside the ECU. Therefore, by monitoring the memory of the ECU during the program execution in the HiL test environment, it is possible to acquire debugging information related to the faults [4, 5].

2.2. Transferring Large Automotive Data Using the CAN Bus. The CAN (Controller Area Network) is a protocol that is commonly used in all ECUs, with a transmission rate from 25 Kbps up to 1 Mbps. However, such a transfer rate is optimized to share small but critical values, such as RPM (revolutions per minute) or temperature values [23]. Thus, in addition to the state information generated while performing conventional operations, another approach is needed to overcome the transmission limitations when transferring to the outside the large amount of data that is also generated during the operation, for example, additional information for debugging. The following techniques have been proposed for using the CAN to transfer big data.

Firstly, methods for improving the CAN protocol’s bandwidth have been proposed. The CAN+ protocol increases the amount of data that can be sent in a message by setting the high-transfer rate in the gray zone existing between regions for synchronizing and sampling the data [9, 10]. FastCAN overcomes the transfer limits by adding an extra bus to the network, which originally consisted of a single bus [11]. Further, it mediates messages via a new medium-access technique, allowing each node to efficiently access the bus. Although both approaches improve the transfer rate by 16 times or more, they cannot be utilized to transfer a large amount of data to the outside in the HiL test environment. When any field in the protocol is modified, except the data field, an overall conversion is required to allow the communication driver to recognize the data in that area. However, since the HiL test proceeds on the assumption that the manufacturing ECU is complete, hardware modifications are not appropriate.
Secondly, reducing the large volume of data by using a compression algorithm has been proposed. Wu et al. presented an algorithm which performs the delta compression to reduce the amount of data when it exceeds 5 bits [12]. Additionally, they suggested another approach to enhance the data compression ratio by rearranging different signals according to their characteristics [13]. In Miucic et al.’s research [14], additional control bits are used to compare the original data to the compressed data in order to send the smaller one. To enhance the algorithm proposed by Miucic, Kelkar suggested the method to diversify the parameters by utilizing the fields that represent the compression information [15]. In advance, the computation is reduced by positioning the compression information in front of the compressed data unlike the above [16]. However, applying such compression algorithms has the following restrictions in the HiL test environment. First, when the compression ratio does not satisfy the communication-bandwidth requirement, the surplus data can be lost. Furthermore, the existing test behavior can be interrupted since the operations for compressing data can be an additional burden on the processor.

Finally, a technique for dividing the data into a size that can be stored in a message has been proposed. The data is divided into segments that can be stored in a message, including the sequence information; then, each segment is transmitted [17]. Using the eight-byte data field in the CAN protocol, the data is stored in six bytes with two bytes of sequence information. In addition, the reserved bits in the control field are used to indicate fragmentation. However, this is not applicable to the HiL test, where a large amount of data is generated periodically inside the ECU. Embedded systems, such as ECUs, have limited storage buffers. Therefore, when monitoring the ECU while the HiL test is in progress and a large amount of memory data is repeatedly generated, there is a possibility that the data will overflow the storage buffer and be lost. Furthermore, when using reserved bits for sequence information, the communication driver must be changed to interpret the modified message, and this cannot be applied to the ECU in the HiL test.

3. Collecting Memory Data for Fault Visualization

This section explains the method for collecting large amounts of memory data periodically by using a data-cascading method in the ECU while executing the HiL test and the algorithm to calculate the memory-update information using the collected data. Moreover, methods for a developer to visualize the update information are suggested.

3.1. Data-Cascading Method for Collecting Extensive Memory Data. To transfer the large amount of memory data generated by the ECU in the HiL test, limitations must be overcome, such as restrictions in utilizing system resources or modifying the hardware. In the data-cascading method, extensive data was successfully transmitted by running a simulation repeatedly with the same test scenario [24]. Figure 2 illustrates the process of collecting the memory data generated inside the ECU using the above method. For example, considering the operation cycle of the system’s main task as 10 ms and the size of the memory to monitor periodically as \( A \times B \), the following process is executed to divide an area into the size of data that can be transferred in each cycle.

First, the number of messages that can be sent in one cycle is calculated by dividing the operation cycle into the time required to send one message successfully. In the example, three messages can be sent in one operation cycle (\( A \times B \), \( A \times B \), and \( A \times B \)). By considering the bus occupancy of the existing operation, it is possible to calculate the number of messages that can be sent without affecting the execution. The size of the area that can be transferred per operation cycle is calculated by multiplying the size of the storables in a message, excluding the sequence information.

In Figure 2, the area to monitor is divided into three regions, and the amount of data that can be sent in one cycle is \( A \). Therefore, to collect the data generated in all of these regions, the simulation must be repeated three times. In the first simulation, the data corresponding to region \( A \) is sent, and in the second simulation, the data corresponding to region \( A \) is sent. This process is repeated until all the data regions are sent to the database.

To collect the data periodically generated in the monitored area, as in Figure 2, the number of divided areas and the simulation repetitions can be calculated using the following information:

(i) \( p_{\text{task}} \): operation cycle of the system’s main task;
(ii) \( p_{\text{msg}} \): minimum time to transfer a message;
(iii) \( N \): number of messages that can be sent in \( p_{\text{task}} \) \( (p_{\text{task}}/p_{\text{msg}}) \);
(iv) \( D_{\text{req}} \): size of data generated in every \( p_{\text{task}} \);
(v) \( D_{\text{msg}} \): size of data stored in a message, excluding the order information;
(vi) \( \text{OCC} \): occupancy of the communication bus for the existing operation.

By considering the existing bus occupancy, the number of messages to send in each cycle is calculated as \( N \times D_{\text{msg}} \times (1 - \text{OCC}) \). Finally, the number of area segments is calculated as dividing \( D_{\text{req}} \) by \( N \times D_{\text{msg}} \times (1 - \text{OCC}) \), as in

\[
\text{# of segments} = \frac{D_{\text{req}}}{N \times D_{\text{msg}} \times (1 - \text{OCC})}.
\]

Meanwhile, the size of the data field in the CAN data frame is eight bytes. To utilize it efficiently, it is necessary to define the proper protocols. Figure 3 shows three protocol types; their detailed explanations follow.

(i) Data-Request Protocol (RP). The data collector in the host PC generates a data-request packet (RP packet) to send the transfer environment and the area to request to the transfer agent in the ECU. The information field is set to 1 to indicate that it is an RP packet. In the rest of the fields, the total number
of frames and the start and end addresses to monitor are included.

(ii) Cascading-Information Protocol (CIP). The transfer agent generates the cascading-information packet (CIP packet) to transfer the sequence information of the data currently transmitted. The information field is set to 1 to indicate that it is the CIP packet. The rest of the fields include the sequence information among the divided regions (offset) and the frame index of the frame to which the currently transmitted data belongs.

(iii) Data-Transfer Protocol (DP). The transfer agent generates a data-transfer packet (DP packet) to transfer the data with the sequence information in the divided regions. The information field is set to 0 to indicate that it is the DP packet. In the rest of the fields, six bytes of successive data is stored, and their starting address is stored in the base address field.

3.2. Computing the Memory-Update Frequency for Fault Visualization. Using the suggested method, the memory data generated in the ECU is collected to acquire the memory-update information. The memory data \( D_{\text{req}} \) is collected in every \( P_{\text{task}} \) during the execution time \( E \). Equations (2) and (3) represent the address range and the frame index.

The frame is defined in (4), which represents the set of data generated in each cycle.

\[
\text{Memory Address Set} (A) \equiv \{ a \mid D_{\text{req,start}} \leq a < D_{\text{req,end}}; a \text{ is an address} \}
\]

\( 2 \)

\[
\text{Frame Index Set} (I) \equiv \{ k \mid 0 \leq k < \frac{E}{P_{\text{task}}}; k \text{ is a frame index} \}
\]

\( 3 \)

\[
\text{Frame} (F_{a,k}) \equiv \{ v_{a,k} \mid \text{Value stored in address } a \text{ generated at time } k \times P_{\text{task}}; a \in A, k \in I \}. \]

The memory-update information is calculated as in (5). By comparing two consecutive frames, the equation assigns 1 when the data has changed and 0 when the data has not changed, or it is the 0th frame.

\[
\text{Memory Update} (U_{a,k}) = \begin{cases} 
0 & \text{if } k = 0 \\
0 & \text{if } v_{a,k-1} = v_{a,k} \\
1 & \text{if } v_{a,k-1} \neq v_{a,k} \\
& (a \in A, k \in I).
\end{cases}
\]

Figure 2: Overall process of moving memory data from the ECU to an external database using a data-cascading algorithm.

Figure 3: Protocols defined in the data-cascading algorithm.
Using the equation above, it is possible to obtain the memory-update information of the $k$th frame. With this information, it is possible to calculate the cumulative update count for each memory address generated during execution, as expressed in

\[
\text{Memory Update Frequency} (\mathcal{UF}_{A,I}) = \sum_{k \in I} U_{a,k} \quad (a \in A).
\]  

### 3.3. Visualizing the Memory-Update Frequency

The collected memory data is stored in frames, as shown in Figure 4(a). Figure 4(a) represents $F_{A,0} - F_{A,3}$, which is the raw data stored in the database. By processing the raw data, $U_{A,1} - U_{A,3}$, $\mathcal{UF}_{A,I}$ can be obtained for visualization as Figures 4(b) and 4(c).

### Visualizing $U_{A,k}$ in Each Time $k$ (Visualizing the Update Information in Each Frame)

$U_{A,k}$ is visualized with the data acquired in (5). Except the 0th frame, the images are generated as the number of frames, and each image contains the information as $U_{A,1} - U_{A,3}$ in Figure 4(b). Each image shows whether an update occurred in each address. An address where no update occurred or randomly occurred can be checked differently to reduce the regions to monitor.

### 4. Design of Modified Architecture in the HiL Test Environment

To collect and visualize the data generated from the ECU while processing the HiL test, the following modules have been designed. Figure 5 depicts the modified architecture from the general HiL test environment. In the host PC, the data collector and visualizer are installed. In the ECU, a transfer agent is installed to perform the functions. The transfer agent in the ECU and the data collector in the host PC are connected to each other, using the CAN to exchange packets. The visualizer generates the images with the data from the database. A detailed explanation of each module follows.

### 4.1. Data Collector

The data collector is installed in the host PC and provides a UI (user interface) to receive the...
### Input:
- CIP - Cascading information packet from transfer agent
- DP - Data transfer packet from transfer agent

### Output:
- \( v_{a,k} \sim v_{a+5,k} \) - Values generated at \( k \)th frame in address \( a \) to \( a + 5 \)

(1) **procedure** RECEIVE_DATA_SEGMENTS
(2) **begin**
(3) \( k \leftarrow \text{CIP.Frame} \)
(4) \( \text{offset} \leftarrow \text{CIP.Offset} \)
(5) **if** DP received **then**
(6) \( a \leftarrow \text{DP.Base} \)
(7) \( \text{data[ ]} \leftarrow \text{DP.Data[ ]} \)
(8) **end if**
(9) **end**
(10) **end procedure**

**Algorithm 1:** Algorithm of the data collector collecting the data.

---

### Figure 5: Modified architecture of the HiL test environment.

The data collector receives the CIP packet and DP packet from the transfer agent and parses them to acquire the data and sequence information to store in the database. Firstly, the offset and the frame index are acquired by analyzing the received CIP packet (lines (3)-(4)). After obtaining the sequence information, the data acquired from the DP packet is stored in the database. From the DP packet, six consecutive bytes of data and their starting address can be acquired (lines (6)-(8)).

Algorithm 2 shows an algorithm for calculating the update information by using the received data. First of all, when the data collector receives the CIP packet, it saves the value of the frame index into \( k \) by paring the packet (line (4)). Then, receiving the following DP packets, it stores the consecutive six bytes of data and their starting address into \( a \) (line (6)). If the value of \( k \) equals 0, it assigns the value 0 from \( U_{a,0} \) to \( U_{a+5,0} \) (lines (7)-(8)). If the value of \( k \) is greater than 0, it compares the received data to the data in the buffer to check the difference. When there is a difference, it assigns 1 to the area where the update occurred (lines (11)-(17)).

### 4.2. Transfer Agent

The transfer agent is implemented as independent software in the ECU; it exchanges packets with the data collector in the host PC by the connection to the CAN. It receives the RP packet from the data collector to obtain the range of memory to monitor and the total frame count. After obtaining the information, it captures the data according to the request, produces the DP packet, and sends it to the data collector. In addition, it sends a CIP packet to transmit the sequence information of the transmitting data. The algorithm for the above functions is depicted in Algorithm 3.

The transfer agent receives the RP packet from the data collector and the environment information, such as \( P_{task}, P_{msg}, D_{req}, D_{msg}, \) and OCC. With the information, the transfer agent operates as follows. Firstly, it obtains the value of the frame index and the start and end addresses of the memory area to monitor from the DP packet (lines (3)-(5)). Then, it divides the memory according to the environment information. The number of regions to divide...
Input:
CIP - Cascading information packet from transfer agent
DP - Data transfer packet from transfer agent
Output:
\[ U_{a,k} \sim U_{a+5,k} \] - The memory-update occurrence from address \( a \) to \( a + 5 \) at \( k \)th frame

(1) procedure UpdateDecision
(2) begin
(3) buffer[]: Temporary buffer for storing the data generated at \( k \)-th frame from address \( a \) to \( a + 5 \)
(4) \( k \leftarrow \text{CIP.Frame} \)
(5) while Receiving data transfer packet do
(6) \( a \leftarrow \text{DP.Base} \)
(7) if \( k = 0 \) then
(8) \( U_{a,k} \leftarrow U_{a+5,k} \leftarrow 0 \)
(9) else
(10) for \( i = 0 \) to 5 do
(11) if buffer[\( i \)] \( \equiv \) DP.Data[\( i \)] then
(12) \( U_{a+i,k} \leftarrow 0 \)
(13) else
(14) \( U_{a+i,k} \leftarrow 1 \)
(15) end if
(16) end for
(17) end if
(18) buffer[] \leftarrow DP.Data[]
(19) end while
(20) return \( U_{a,k} \sim U_{a+5,k} \)
(21) end
(22) end procedure

Algorithm 2: Algorithm of the data collector operating the update decision.

can be calculated as shown in Algorithm 4, which requires the values of \( P_{\text{req}} \), \( P_{\text{msg}} \), \( D_{\text{req}} \), \( D_{\text{msg}} \), and OCC (Algorithm 4, lines (3)-(4)). When the calculation finishes, it generates the CIP packet using the current frame index and the offset value and the DP packets included in the above sequence information (lines (10)-(13)). The packets are sent to the data collector; this process is repeated while increasing the offset and frame index.

4.3. Visualizer. The visualizer generates images to display the data stored in the database. Two types of images are generated in the visualizer. First, it generates a list of images containing the updated information in each address in each frame. It uses \( U_{A,k} \), calculated using (5), and generates the images for the number of frames to check the update information in every cycle. Areas where no updates occurred or areas unrelated to the operation can be filtered out. Second, it creates an image containing the total memory-information count in each address. It generates one image using \( UF_{A,1} \), calculated using (6). The image displays the update frequency in different colors to provide an intuitive understanding.

5. Experimental Results

5.1. Memory Data Acquisition. For the implementation, we appended the transfer agent, data collector, and the visualizer modules into the HiL test environment, as shown in Figure 5. The transfer agent and the data collector are installed in the ECU and the host PC to operate the algorithms in Section 4. The environmental information for the dividing memory regions is as follows:

(i) \( P_{\text{task}} \): 10 ms;
(ii) \( P_{\text{msg}} \): 300 \( \mu \)s;
(iii) \( N = P_{\text{task}}/P_{\text{msg}} = (10 \times 10^{-3})/(300 \times 10^{-6}) = 33.33 \approx 33 \);
(iv) \( E = 4 \) s (\( \# \) of frames = 4 s/10 ms = 400);
(v) OCC: 55%;
(vi) \( D_{\text{msg}} \): 6 bytes;
(vii) \( D_{\text{req.start}} \): 0;
(viii) \( D_{\text{req.end}} \): 29,400;
(ix) Lab ID: 74.

In the environment above, \( D_{\text{req}} \) is 29,400 bytes, which means that the amount of data generated in every \( P_{\text{task}} \), which is 10 ms, is 29,400 bytes. Since CAN’s maximal bandwidth is about 1Mbps, while the data generated is about 3MB per second, this causes a loss of about 96% of the data in each cycle. The number of memory regions to divide was calculated as 330, using (7). Therefore, we repeated the
### Algorithm 3: Algorithm of the transfer agent.

```plaintext
procedure SendDataSegments
begin
    F ← R.P.Frame
    a_start ← R.P.StartAddress
    a_end ← R.P.EndAddress
    x, y ← 0
    D_req ← a_end − a_start
    Seq ← CalculateSegments(P_task, P_msg, D_req, D_msg, OCC)

    while y < Seq do
        while x < F do
            CIP.Frame ← x
            CIP.Offset ← y
            DP.Base ← the address corresponding to offset y
            DP.data[ ] ← 6 bytes of successive data starting from
                the address of DP.Base
            Send CIP and DP to the data collector
            x ← x + 1
        end while
        x ← 0, y ← y + 1
    end while
end procedure
```

In the results shown in Figure 6(a), 400 frames of data, about 11,760,400 bytes, were collected successfully. About 85,956 updates occurred during the program execution; they were calculated using (5).

To verify the collected data, we used the Trace 32 hardware debugger [25]. Trace 32 is a tool that controls and monitors the target through a standard debug interface, such as JTAG or BDM, equipped inside the processor. It is possible to collect the variables, registers, and OS information by suspending the program during execution. However, there is a limit to collecting memory data during the HiL test because this process must stop the execution during the simulation. Therefore, to utilize Trace 32 for verification, we stopped the execution randomly and captured the data stored in the stack, which is a part of memory. We compared the captured stack with the data collected by the proposed method.

![Figure 6: Data collection results and the total memory-update count from the HiL test.](image-url)
Input:
- \( P_{\text{task}} \) - Operation cycle of system main task
- \( P_{\text{msg}} \) - The time required for sending one CAN message
- \( D_{\text{req}} \) - The size of memory to monitor
- \( D_{\text{msg}} \) - The size of data can be stored in one CAN message
- \( \text{OCC} \) - CAN bus occupancy of the existing operation

Output:
- \( \text{Seg} \) - the number of segments to slice

(1) function \( \text{CalculatingSegments}(P_{\text{task}}, P_{\text{msg}}, D_{\text{req}}, D_{\text{msg}}, \text{OCC}) \) begin
(3) \( N \leftarrow \frac{D_{\text{msg}}}{P_{\text{task}}} \)
(4) \( \text{Seg} \leftarrow \frac{D_{\text{req}}}{N \times \text{OCC} \times D_{\text{msg}}} \)
(5) return \( \text{Seg} \)
(6) end
(7) end function

Algorithm 4: Function to calculate the number of segments.

Figure 7: Captured data in a stack randomly using Trace 32 and the data collected using proposed algorithm.

Figure 7(a) is the stack and register view; it displays the stack and the stack pointer captured at any time during the execution by Trace 32. On the left side of Figure 7(a), register R1 stores the value of the stack pointer, which is the starting address of the stack, and the right side of Figure 7(a) shows an image of the stack. Starting from address 40006BA4, data is stored on the stack every four bytes. Figure 7(b) is the set of data starting from 40006BA4 collected by the proposed method. We compared the data captured by Trace32 and the data collected by our method and found that they were identical. This means that the data collected by our method can be reliably used.

5.2. Application. To visualize the data collected by the method proposed in this paper, we used the MFC (Microsoft Foundation Class) framework and the VTK (Visualization
Tool-Kit) library, which is open source based on OpenGL [26]. In addition, we utilized the GNU binary utilities to acquire information such as the stack or section, which can be generated by static analysis [27]. Using this information can reduce the area to check for debugging by filtering out unnecessary information, based on the symbol or section information mapped at each address and the memory-update frequency. Using the symbol information, we can distinguish the variables that are not necessary to understand the operation, such as the OS stack for which the memory-update frequency is meaningless or the buffer and timer which are used dynamically [5]. Therefore, the debugging time can be shortened by reducing the fault candidates in accordance with the symbol information. With the processed data mentioned above, it is possible to create two image types using the visualization methods proposed in this paper.

*Visualizing $U_{FAJ}$ (Visualizing Total Update Frequency).* Figure 8 shows an image containing the total update frequency during the program execution for each address. Each value has a different color according to the number of updates. The higher number of updates has a color close to purple and red, and the lower number of updates has a color close to green and yellow. The areas where no updates occurred are white. The candidates to check for debugging can be reduced by filtering out the variables that have no relevance to the operation or where the update frequency is meaningless.

In Figure 8, $\odot$ is related to the stack used in the OS, $\ominus$ is the buffer area, and $\bullet$ is the counter and timer area. Each area has characteristics, such as having dynamic values or being temporarily used as buffers, and the stored data value has meaning while its update information is meaningless. Therefore, these areas can be represented as white to reduce the fault candidates. By using the image in Figure 8, developers can select the focus area based on the total update frequency related to the operation.

*Visualizing $U_{A,k}$ in Each Time $k$ (Visualizing Update Information in Each Frame).* The memory-update information in each frame is visualized in Figure 9. The left side of the figure is a list of images containing the memory-update information that occurred in every cycle of the system’s main
task during the execution. Each image in the list can be expanded to the right side of the image. Gray means that it has no relevance to the operation or no updates that occurred during the execution. Red means an update that occurred in the selected time. White means that an update that occurred during execution, but not in the selected frame. With the list of images, a developer can understand the memory-update phenomenon during execution in the area from Figure 8 that the developer selected to focus. Using this information, the developer can grasp when a fault occurred. The data stored in each frame is also provided in the form of a table.

With the utilization above, developers have the following advantages for understanding the causes of faults generated during the HiL test. First, it is possible to catch both the time point when the fault occurred, using the memory-update information, and also visual information, such as the combination of symbols causing the faults, in environments where source code use is restricted, such as black-box testing. In addition, since the size of the memory data collected periodically is large, it is more effective to intuitively access the cause of the faults using images such as Figures 8 and 9 rather than that provided in the text.

6. Conclusion

In this paper, we proposed a method to collect large amounts of memory data generated inside the ECU by using CAN to monitor the behavior in the HiL test environment, where using a general debugger, utilizing additional system resources, and modifying the hardware are restricted. In the proposed method, we divided the memory into a size that could be sent during the operation cycle, considering the communication environment. Then, we repeated the simulation with the same test script to collect the entire data region, and the collected data was verified. In addition, we provided useful information to access the cause of faults by visualizing data obtained from the memory-update information.

A large amount of data could be transmitted using our method, which does not require hardware modifications or using a debugger and also does not affect the existing operation. Furthermore, in the HiL test environment, where using source code is limited, effective access to the fault was possible by utilizing the images containing the memory-update information. As a result, the debugging time was shortened, compared to finding the faults using text.

For future work, we will dynamically consider the bus occupancy to fully utilize the bus to enhance the transmitting efficiency. With the enhanced efficiency, the number of simulation repetitions can be reduced. Moreover, a more intuitive understanding can be provided when the symbol and section information assigned to each memory address is provided together with the images.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the MSIP (NRF-2016R1A2B1014376).

References

[1] P. Braun, M. Broy, F. Houdet et al., “Guiding requirements engineering for software-intensive embedded systems in the automotive industry: the REMsES approach,” Computer Science - Research and Development, vol. 29, no. 1, pp. 21–43, 2014.
[2] E. Bringmann and A. Krämer, “Model-based testing of automotive systems,” in Proceedings of the 1st International Conference on Software Testing, Verification and Validation (ICST ’08), pp. 485–493, April 2008.
[3] M. Gomez, “Hardware-in-the-loop simulation,” Embedded Systems Programming, vol. 14, pp. 38–49, 2008.
[4] F. Vahid and T. Givargis, Embedded System Design: A Unified Hardware/Software Approach, Department of Computer Science and Engineering, University of California, 1999.
[5] K. Choi, J. Seo, S. Jang, and J. Lee, “HiL test based fault localization method using memory update frequency,” in Advances in Computer Science and Ubiquitous Computing, vol. 373 of Lecture Notes in Electrical Engineering, pp. 765–772, Springer, Singapore, 2015.
[6] S. Tuohy, M. Giffin, C. Hughes, E. Jones, M. Trivedi, and L. Kilmarin, “Intra-Vehicle Networks: A Review,” IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 2, pp. 534–545, 2015.
[7] N. Navet, Y. Song, F. Simonot-Lion, and C. Wilwert, “Trends in automotive communication systems,” Proceedings of the IEEE, vol. 93, no. 6, pp. 1204–1222, 2005.
[8] A. Sangiovanni-Vincentelli and M. Di Natale, “Embedded system design for automotive applications,” Computer, vol. 40, no. 10, pp. 42–51, 2007.
[9] T. Ziermann, S. Wildermann, and J. Teich, “CAN+: a new backward-compatible controller area network (CAN) protocol with up to 16x higher data rates,” in Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE ’09), pp. 1088–1093, Nice, France, April 2009.
[10] T. Ziermann and J. Teich, “Electromagnetic Compatibility (EMC) of CAN+,” in GMM-Fachbericht-AmE 2010—Automotive meets Electronics, VDE, 2010.
[11] G. Cena and A. Valenzano, “FastCAN: a high-performance enhanced CAN-like network,” IEEE Transactions on Industrial Electronics, vol. 47, no. 4, pp. 951–963, 2000.
[12] Y. Wu, J.-G. Chung, and M. H. Sunwoo, “Design and implementation of CAN data compression algorithm,” in Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS ’74), pp. 582–585, IEEE, Victoria, Canada, June 2014.
[13] Y.-J. Wu and J.-G. Chung, “Efficient controller area network data compression for automobile applications,” Frontiers of Information Technology & Electronic Engineering, vol. 16, no. 1, pp. 70–78, 2016.
[14] R. Miuic, S. M. Mahmud, and Z. Popovic, “An enhanced data-reduction algorithm for event-triggered networks,” IEEE Transactions on Vehicular Technology, vol. 58, no. 6, pp. 2663–2678, 2009.
[15] S. Kelkar and Rajkamal, “Control area network based quotient remainder compression-algorithm for automotive applications,” in Proceedings of the 38th Annual Conference on
IEEE Industrial Electronics Society (IECON ’12), pp. 3030–3036, October 2012.

[16] S. Kelkar and R. Kamal, “Boundary of fifteen compression algorithm for controller area network based automotive applications,” in Proceedings of the International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA ’14), pp. 162–167, Maharashtra, India, April 2014.

[17] C. Shin, “A framework for fragmenting/reconstituting data frame in Controller Area Network (CAN),” in Proceedings of the 16th International Conference on Advanced Communication Technology (ICACT ’14), pp. 1261–1264, February 2014.

[18] R. Oshana, Software Engineering for Embedded Systems: Methods, Practical Techniques, and Applications, Newnes, Oxford, UK, 2013.

[19] S. P. Reiss, “Visual representations of executing programs,” Journal of Visual Languages and Computing, vol. 18, no. 2, pp. 126–148, 2007.

[20] J. A. Jones, M. J. Harrold, and J. Stasko, “Visualization of test information to assist fault localization,” in Proceedings of the 24th International Conference on Software Engineering (ICSE ’02), pp. 467–477, ACM, Orlando, Fla, USA, May 2002.

[21] H. Shokry and M. Hinchey, “Model-based verification of embedded software,” Computer, vol. 42, no. 4, pp. 53–59, 2009.

[22] J. S. Keränen and T. D. Rätty, “Model-based testing of embedded systems in hardware in the loop environment,” IET Software, vol. 6, no. 4, pp. 364–376, 2012.

[23] S. Corrigan, “Texas instrument,” Application Report, 2008.

[24] J. Lee, K. Choi, S. Y. Jang, and J. Lee, “Data cascading method for the large automotive data acquisition beyond the CAN bandwidth in HiL testing,” in Advances in Computer Science and Ubiquitous Computing, Lecture Notes in Electrical Engineering, pp. 773–780, Springer, Singapore, 2015.

[25] Trace 32, http://www.lauterbach.com/frames.html?debugger features.html.

[26] VTK-The Visualization Toolkit, http://www.vtk.org/.

[27] GNU Binary Utilities, https://sourceware.org/binutils/docs/binutils/index.html.
Research Article

Efficient Thread Mapping for Heterogeneous Multicore IoT Systems

Thomas Mezmur Birhanu,1 Zhetao Li,2 Hiroo Sekiya,3 Nobuyoshi Komuro,3 and Young-June Choi1

1Department of Computer Engineering, Ajou University, Suwon 16499, Republic of Korea
2College of Information Engineering, Xiangtan University, Hunan 411105, China
3Graduate School of Advanced Integration Science, Chiba University, Chiba 263-8522, Japan

Correspondence should be addressed to Young-June Choi; choiyj@ajou.ac.kr

Received 17 November 2016; Accepted 16 January 2017; Published 27 February 2017

Academic Editor: Jeongyeup Paek

Copyright © 2017 Thomas Mezmur Birhanu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a thread scheduling mechanism primed for heterogeneously configured multicore systems. Our approach considers CPU utilization for mapping running threads with the appropriate core that can potentially deliver the actual needed capacity. The paper also introduces a mapping algorithm that is able to map threads to cores in an $O(N \log M)$ time complexity, where $N$ is the number of cores and $M$ is the number of types of cores. In addition to that we also introduced a method of profiling heterogeneous architectures based on the discrepancy between the performances of individual cores. Our heterogeneity aware scheduler was able to speed up processing by 52.62% and save power by 2.22% as compared to the CFS scheduler that is a default in Linux systems.

1. Introduction

The multicore architecture has penetrated into the industry because of its apparent advantage in aiding for higher performance. And more recently, a heterogeneous variant of this architecture has also found its way into the mainstream ARM’s big.LITTLE [1] for various Internet-of-Things (IoT) applications as well as mobile devices. The popularity of big.LITTLE architectures is found in their better performance-power ratio and more efficient space utilization than their homogeneous counterparts [2]. These architectures consist of cores having different computing capacities but under the same instruction set architecture (ISA).

They act as a way to conform to the variation witnessed in a typical workload sets, thereby helping maintain performance while saving power through aversion of core underutilization. Other than underutilization, harboring a fast thread on sluggish core would be penalized by a significant power consumption since it would take a long time for it to complete, as seen in [3]. The variation between core performance and energy footprint is the result of discrepancies between their features like cache size, frequency, issue width, in-order/out-order, and such [4–9]. Though this multicore architectures seem to be quite advantageous, designing an app littered to a multicore, let alone heterogeneous, architecture is made difficult because the developer is usually exposed to the low level details. Paper [10], proposes a developer friendly interface to harness GPUs and other multicore architectures. As such, we believe much of the logic for harnessing these architectures should be pushed to the OS scheduling. In [11], they even went as far as proposing their own experimental heterogeneous programming system.

These heterogeneous architectures are a new trend; as such they can render previously existent schedulers obsolete. The Completely Fair Scheduler (CFS) in Linux is one of them. CFS was introduced starting form Linux 2.6.23 as a replacement of the previous vanilla scheduler’s SCHED_OTHER interactivity code [12]. CFS is widely chosen against other schedulers such as FIFO, in that it averts the problem of process starvation and assures fairness. CFS does not account
core heterogeneity while making its scheduling decision. This can bear a far reaching consequence in terms of the energy and performance that can potentially be saved and improved, respectively. As CFS is the default scheduler in Linux, our focus will be towards its improvements in the arena of heterogeneous architectures.

This paper proposes Fastest-Thread-Fastest-Core (FTFC) dynamic thread scheduling mechanism that bases its mapping decision on conformity of running threads CPU utilization with the performance of available cores. It does this by dynamically and periodically measuring the CPU utilization of running threads so those threads that exhibit high CPU utilization are assigned to cores that can deliver high performance needed, while those threads with low CPU utilization are assigned to low performance cores. We measure the threads CPU utilization based on their CPU time; that is, the amount of time that thread spends executing on the core, as opposed to, for example, waiting for input/output (I/O) operations or entering low-power (idle) mode. The FTFC scheduler does its mapping using a new scheme which we coined as Binary Searched Mapping (BSM), which runs in \( O(N \log M) \) time complexity, where \( N \) is the number of cores and \( M \) is the number of type of cores.

This paper also introduces a new means of modeling heterogeneity based on the discrepancy between the performances of cores. We coined it as heterogeneity measure (HM) that describes how heterogeneous the system is. Using this measurement technique we are able to consider a whole spectrum of heterogeneous configurations. Through our experimentation, the FTFC scheduler saves power by up to 2.22% and speeds up processing by up to 52.62%, compared to CFS.

The rest of the paper is organized as follows. Section 2 elaborates on related works. Section 3 presents our heterogeneity measure. Section 4 describes the proposed scheduler. Section 5 shows the experimental environment and results. Finally, Section 6 concludes.

2. Related Work

Regarding multicore scheduling, several studies are found in literature. In [13], Koufaty et al. proposed a method of looking at the CPI stacks of the running applications to infer the core to which they would belong to. The HASS technique, proposed by Shelepov et al. [14], sought for efficient mapping by using program signatures which are embedded in the program binaries prior to execution. In [15], Chen and John used Weighted Euclidean Distance (WED) measure between the desired resource demand of a program and the core configurations to map a thread to an optimal core. Both techniques rely on prior offline profiling of the programs.

Most previous work relied on analysis of programs into their constitute subtasks, formulating constraint graph, and partitioning the graph to minimize the overall execution time [16] or the data transfer on buses [16, 17]. Their approach requires the microarchitecture dependent performance/power data as a priori information for scheduling, whereas our approach does not need this. Chen and John [18] used fuzzy-logic based program-core suitability computation. This method is inapplicable when the number of core characteristics increases. This is because the complexity of their algorithm increases exponentially as the characteristics increases. Our approach is totally oblivious to the number or type of characteristics, as it only concerns the core performance.

Gulati et al. [19] explored the benefits exhibited from dynamic mapping of tasks to cores of flexible design. Kumar et al. [2] proposed a dynamic program scheduling scheme based on sampled EDP in subsequent runs. This is just a straightforward trial-and-error scheduling approach to find the appropriate match of core and thread. Becchi and Crowley [20] followed up on their work by using IPC ratios between two programs for program migrations. All these methods, as do ours, exploit inaprogram diversity and could adapt to program phase changes. In [21], a dynamic scheduling was proposed by tweaking core priority so as the OS load balancer does its job in a biased manner. This bias is formulated by taking identify key metrics.

The work in [22] proposes a static heuristic method for heterogeneous processors based on the use of an acyclic precedence graph. The use of this heuristic, however, requires a priori knowledge of the characteristics of the workloads. One of relatively recent papers that seek to address this mapping problem is investigated by Liu et al. [23]. They mathematically formulated the problem as a 0-1 integer linear program (Integer LP) and also proposed an iterative \( O(N^2/M) \) heuristic algorithm for solving it. We were able to do it in an \( O(N \log M) \) fashion.

We assume that each core is executing a single thread at most and assumed that the variations of the workloads rather than the type of workloads are more important when choosing the type of benchmarks. In [24], they refer to the variation of workload as more important metric when they designed their flexible multicore (FMC) architecture. Reference [25] advocates individual cores properties to be tailored to the different subsets of application characteristics as a main idea behind their designs.

3. Heterogeneity Measure

With the introduction of heterogeneity to architectures, the need to not only tell the heterogeneity but also the degree of the heterogeneity itself became apparent. We formulated it as the standard deviation between the normalized performances of individual cores. The performance of cores is once again modeled by total CPU utilization gotten from running four of PARSEC benchmarks serially on that core, core performance being directly proportional to the average CPU utilization. We define heterogeneity measure as

\[
HM = \sqrt{\frac{\sum_{i=1}^{N} \left( T_{i} - T_{\text{avg}} \right)^2}{T_{\text{max}}}},
\]

where \( T_{i} \) is the \( i \)th core CPU performance. From the equation above we can deduce that if we had a heterogeneous dual core configuration and the variance of 0.5 (\( HM = 1/\sqrt{2} \)), it
means one core is twice as fast as the other, where $T_{\text{max}}$ is one of either $T_1$ or $T_2$. HM can never be greater than or equal to 1. If it is 1, it means only one core is functioning. We use HM as an important metric to express heterogeneity of the configurations we used throughout the experimentation.

4. Fastest-Thread-Fastest-Core Scheduler

Unlike heuristic mapping methods proposed in [23], our method does not need to model the power, as it bases its methods based on the understanding of power consumption being dependent upon the independent variable performance. As such, we focus on performance optimization and let the dependent variable power follow up with its own optimization.

Before the FTFC scheduler takes over, six of the PARSEC benchmarks are launched and randomly assigned to one of the six available cores. Stage 1 of Figure 2 shows an example how it is done using four cores and four threads. $T$ signifies the normalized CPU performance. $U$ signifies the throughput of the threads on their respective core. The lifetimes of these threads are assigned arbitrary, the maximum being 30 seconds and the minimum being 5 seconds. Each thread was assigned the same priority as the other. The scheduling process ends as all threads end their execution. CPU utilization was monitored by measuring the CPU time. FTFC is the same as CFS except that FTFC considers heterogeneity and does dynamic thread migrations. When the configuration is homogeneous, that is, HM = 0, FTFC is essentially CFS, as there will not be any thread migrations. In such ways, FTFC can be seen as an extension of CFS.

Once all the threads are executed, the FTFC scheduler periodically monitors the CPU utilizations of the running threads. We set the time interval to be 1 ms. For such granularity, previous work [26] has shown cold cache effect and context migration costs can be brought to minimum. Some of the considerations that should be taken when doing migrations between cores with private L1s and L2s and shared L3 are, remembering the predictor state, allowing L2-L2 data transfers from inactive to active core and also maintaining the tags of the various L2 cache coherent. Transferring between registers and flushing a dirty private L1 data have a negligible performance overhead [26]. The FTFC scheduler can endure at most $N$ migrations, where $N$ is the number of cores. The diagrammatic illustration of FTFC scheduler is shown on Figure 1.

![Flow diagram of FTFC scheduler.](image-url)
4.1. Homogenization. Once the CPU utilization of a thread is gotten, it is homogenized. This process is necessary as it helps avoiding unfair comparison of CPU utilizations between threads running on small cores and those on bigger cores. “Small” and “Big” describe the relative processing power of the cores. By homogenizing we get the core-neutral CPU utilization value of the threads. It is done by multiplying the CPU utilization of a thread with a ratio of the normalized performance of the fastest core with the core it is running on \( T_{\max}/T_i = 1/T_i \). The homogenizing process is expressed as
\[
U_i = \frac{U_i}{T_i},
\]
where \( T_i \) is the normalized \( i \)th core performance and \( U_i \) is the CPU utilization of \( i \)th thread. Stage 2 of Figure 2 shows the homogenizing process. We can see that thread 1 has same core utilization as thread 4 on stage 1. After the threads are homogenized thread 4 has lower utilization per core than thread 1. This means thread 4 was using less of the performance that core 1 could potentially deliver. The homogenizing process is needed to avoid static comparison of utilization values between cores, which is shown in the scenario of thread 1 and thread 4. The homogenization gives fairness in comparison of threads utilization according to the relative performance of cores.

4.2. Normalization. The next step is to normalize the homogenized CPU utilization of the running sets of threads. By normalizing we know exactly which thread is the most CPU utilizing among the whole sets of threads, because we already homogenized it.
\[
U_i = \frac{U_i}{U_{\max}},
\]
where \( U_i \) is the \( i \)th threads CPU utilization and \( U_{\max} \) is the maximum CPU utilization among the threads.
(1) //Individual core types performance is already gotten and listed in ascending order
(2) Let \( W_i \) be the \( i \)th core performance
(3) Let \( P_j \) be the \( j \)th thread in thread queue (\( j = 1 \cdots L \))
(4) Let \( C_i \) be the processor of core type \( i \) (\( i = 1 \cdots M \))
(5) for \( j (1 \cdots L) \)
(6) \( T_j = j \)th thread CPU utilization
(7) \( left = 0 \)
(8) \( right = \) number of type of cores
(9) while right \( ! = \) left
(10) \( mid = (right + left)/2 \)
(11) if \( W_{mid} \) > \( T_p \)
(12) \( right = \) mid
(13) else
(14) \( left = \) mid
(15) end if
(16) end while
(17) place thread \( T_j \) on core \( C_{mid} \)
(18) end for

**Algorithm 1: BSM algorithm.**

Using this, we map the relatively high utilizing thread to the relatively high performance core, and the relatively low utilizing thread to the relatively low performance core. Using such relativity technique we ensure a complete abstraction of the hardware details of the core and program characteristics of a thread into a single variable of CPU performance and utilization value, respectively. In contrast to Performance Impact Estimation (PIE) proposed in [27], FTFC simplifies this estimation greatly. The relativistic nature of this mapping also ensures the avoidance of underutilization in a better way as compared to, for example, Faster-Core-First Scheduling as seen in [28], which aggressively assigns threads to the fastest core. If the core to be migrated upon is empty, the thread is simply migrated but if not we do the migration based on the to be migrated thread’s utilization that is closest to CPU performance as compared to that of the already resident threads.

Stage 3 of Figure 2 shows the normalizing process. Here, the thread utilization is normalized primarily to binary search it through an already sorted list of normalized core performances.

### 4.3. Binary Searched Mapping

After the homogenizing and normalizing phases, the mapping phase takes over. Here, we propose to use binary searching mapping (BSM) to reduce time complexity and computation overhead. The mapping of threads is performed in a binary search manner and the already sorted list of CPU performance is searched for the CPU utilization value of the subject thread. This would ensure that the final entry at which searching terminates is the closest CPU performance to the threads utilization in question. This is done from lines 6 to 15 of Algorithm 1. This searching runs at \( O(\log M) \), where \( M \) is the number of unique cores, and, in the worst case, that is, if all the cores except the last one of the core types are occupied, it takes \( N \) times of iteration to reach it. BSM ensures a complexity of \( O(N \log M) \) where \( M \) is the number of distinct core types in the system and \( N \) is the number of threads to be mapped.

Though the scope of this paper is limited to heterogeneous multicore architectures in mobile devices, for the purpose of validating our mathematically gotten time complexity, we created dummy thread performance values and mapped them to a set of dummy cores but with the same HM always. We generated a set of cores ranging in numbers from 2 to 1000 and observed the trend of the postulated time complexity to the actual observed time. The results of this experiment can be seen in Figure 3. These findings can also suggest the significance of the BSM algorithm in an expanded scope of heterogeneous multicore processors as a whole.

As can be seen in Table 1, the BSM algorithm with the FTFC scheduler outperforms the others. The WED algorithm requires the offline profiling of applications based on their core complementary characteristics, which is another encumbrance to the compiling process and a loom on its scalability. The naive approach to this mapping problem runs at \( O(N!) \) and efficient mapping is guaranteed (mapping wise), but it is too computationally expensive to be used, especially in resource constrained mobile devices. MTS runs at \( O(N^2/M) \). The main drawback of the MTS heuristic approach is that it does not completely exhaust all possible swaps before it terminates. Its swaps are defined and prioritized by the ratio of the change in power to the change in throughput that the swap could produce. This leaves out combinations of swaps that could be possibly valuable, thus not always resulting in optimized mapping.

| Algorithm                              | Time complexity     |
|---------------------------------------|---------------------|
| Binary searched mapping (BSM)         | \( O(N \log M) \)   |
| Weighted Euclidean distance (WED)     | \( O(NM) \)         |
| Maximize then swap (MTS)              | \( O(N^2/M) \)      |
| Naive                                 | \( O(N!) \)         |

**Table 1: Thread mapping time complexity.**
Table 2: Heterogeneous core configurations.

| Heterogeneity | Core | Issue width | L2 cache | Branch predict size | ROB size |
|---------------|------|-------------|----------|---------------------|---------|
| 0             | 0    | 4           | 512 KB   | 16384               | 256     |
|               | 1    | 4           | 512 KB   | 16384               | 256     |
| 0.02          | 0    | 64          | 1 MB     | 1                   | 4096    |
|               | 1    | 4           | 512 KB   | 265                 | 64      |
| 0.04          | 0    | 64          | 4 MB     | 2048                | 2048    |
|               | 1    | 64          | 128 KB   | 4098                | 1024    |
| 0.06          | 0    | 4           | 1 MB     | 2048                | 64      |
|               | 1    | 64          | 4 MB     | 512                 | 64      |
| 0.08          | 0    | 16          | 4 MB     | 256                 | 32      |
|               | 1    | 64          | 2 MB     | 256                 | 32      |
| 0.1           | 0    | 64          | 4 MB     | 16384               | 1024    |
|               | 1    | 1           | 256 KB   | 1024                | 512     |

5. Results

To conduct our experiments, we used ESESC [29] simulator which supports the heterogeneous configurations and the multitudinousness we needed. The ESESC (enhanced SESC) simulator uses Time Based Sampling (TBS), which they deemed to be ideal when working with multithreaded executions. As seen in Figure 1, the workloads we used for the experiment consist of six PARSEC benchmarks [30], namely, swaptions, canneal, fluidanimate, lu_cb, blackscholes, and freqmine. All these benchmarks were used in their medium sized inputs. We had to change these benchmarks into libraries prior to our simulation since ESESC supports executing one program at a time that can run independent of each other on separate cores. We had two binaries, one for FTFC and the other for CFS. These schedulers take turns to run and spawn all the six benchmarks from the PARSEC libraries they included, during which their power consumption and throughput are monitored and stored. One thread is allowed to execute for a maximum of 30 seconds and a minimum of 5 seconds. We took 30 samples in this manner and compared the difference.

We define core heterogeneity as any inconsistencies between complementary features of cores, such as microarchitectures, frequency/voltage, or any attributes that can cause performance differences. One of the important program characteristics are instruction level parallelism (ILP), data locality, and branch predictability. The respective attributes for cores are Issue Width, Data Cache Size, and Branch Predictor Size. As such, we achieved core heterogeneity based on variations of these three attributes in addition to the ROB Size. We assume that these four attributes produce significant core performance discrepancy enough to create the arbitrary heterogeneity needed. All other attributes, including the frequency and voltage, were maintained constant. Voltage and frequency were kept constant for each core type. The ESESC simulator allows the modification of various attributes of the cores except the frequency or voltage. As such, there was no problem maintaining these, as it runs on their default static configurations.

We assume each core is executing a single thread at most and assumed the variations of the workloads rather than the type of workloads are more important when choosing the type of benchmarks. Our experiments contained two types of cores sharing a common L3 cache of size 3 Mb and having a dedicated L2 cache of the same or different size. There were three caches of each core type.

To create the configuration of required heterogeneity, we first made a core generator script whose sole purpose was to create a set of cores with random variations of their prominent characteristics. We generated 2000 unique core types by randomly varying the Issue Width, L2 Cache, Branch Predict size, and ROB size. Issue width is 4th power from the range of 1 to 64. L2 Cache is a power of 2 ranging from 128 to 32768. Branch Predict and ROB size are a power of 2 from the range of 16 to 32768. This is unlike the method used in [31], where the instruction set of homogeneous processors is augmented via custom instructions.

The second phase was to assess the performance of generated cores. We did this by running four of the PARSEC benchmarks serially on each core type for a total of 60 seconds. We profiled each core type by calculating the total CPU utilization achieved by those running benchmarks. The benchmark selection is based on their proximity to the characteristics of workloads exhibited in mobile devices. The benchmarks we used for this purpose were swaptions, canneal, fluidanimate, and freqmine.

The third step was to select cores in such a way as to guarantee their combination results in the heterogeneity desired. We took those 2000 utilization values from the second step and used them to generate a set of cores that satisfy any heterogeneity we desired but under the roof of heterogeneity possible given the set of variations in the recorded utilizations. We generated six configurations with an incremental heterogeneity of 0.02, starting from HM = 0. We run six of the PARSEC benchmarks on these cores and used Linux’s CFS and the proposed FTFC to schedule them and recorded the total power and throughput. All the six core configurations used can be seen in Table 2.

We compared the result of FTFC with that of CFS. We can see from the result in Figure 4 that when the system is in its
highest configuration, that is, \( HM = 0.1 \), FTFC saves power by 2.22% and improves performance by 52.62% as compared to CFS scheduler. Figure 5 shows the performance of FTFC as compared to CFS. The variations of workload utilization also play an important role for FTFC scheduler operation. We can see from Figures 4 and 5 that the power and throughput for FTFC are essentially the same as CFS when the system is in its homogeneous configuration; that is, \( HM = 0 \). This is because FTFC is built on top of CFS and zero heterogeneity reverts it to operate as CFS. The highest performance gain is gotten when the system was in its highest configuration. The FTFC generally maintains its advantage over the CFS in both its power and throughput across the heterogeneous configurations we investigated.

6. Conclusion

This paper presented a solution to thread mapping on heterogeneous configurations using thread-core performance complementarity based on their CPU utilization and core capacity. Core performance was modeled by getting the total CPU utilization of fixed set of benchmarks running on a core. We were able to achieve a power save of 2.22% and a performance ramp up of 52.62% compared to the CFS scheduler. We were able to do this with an improved time complexity of \( O(N \log M) \). As future work, we propose to include frequency/voltage to achieve higher heterogeneity measures.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research was supported by the MISP (Ministry of Science, ICT & Future Planning), Korea, under the National Program for Excellence in SW supervised by the IITP (Institute for Information & communications Technology Promotion) (R2215161002010002) and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2015R1D1A1A01058025) and supported by the Ajou University Research Fund.

References

[1] P. Greenhalgh, *Big. Little Processing with ARM Cortex-A15 & Cortex A7*, White Paper, ARM, Boston, Mass, USA, 2011.
[2] R. Kumar, K. I. Farkas, N. P. Jouppi, P. Ranganathan, and D. M. Tullsen, “Single-ISA heterogeneous multi-core architectures: the potential for processor power reduction,” in *Proceedings of the 36th International Symposium on Microarchitecture (MICRO ’03)*, pp. 81–92, IEEE, San Diego, Calif, USA, December 2003.
[3] S. Kim, H. Kim, J. Kim, J. Lee, and E. Seo, “Empirical analysis of power management schemes for multi-core smartphones,” in *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication (ICUIMC ’13)*, Kota Kinabalu, Malaysia, January 2013.
[4] K. Asanovic, R. Bodik, B. Catanzaro, J. Gebis, and P. Husbands, “The landscape of parallel computing research,” A View from Berkeley, UC Berkeley Technical Report/Electrical Engineering and Computer Sciences 2006-183, 2006.
[5] S. Borkar, “Thousand core chips—a technology perspective,” in *Proceedings of the 44th ACM/IEEE Design Automation Conference (DAC ’07)*, pp. 746–749, June 2007.
[6] S. Smith, “Many-core operating systems,” in *Proceedings of the Workshop on the Interaction between Operating Systems and Computer Architecture*, in conjunction with ISCA-34, Keynote Speech, 2007.
[7] Q. Han, M. Fan, O. Bai, S. Ren, and G. Quan, “Temperature-constrained feasibility analysis for multicore scheduling,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 12, pp. 2082–2092, 2016.
[8] G. Yao, R. Pellizzoni, S. Bak, H. Yun, and M. Caccamo, “Global real-time memory-centric scheduling for multicore systems,” Institute of Electrical and Electronics Engineers. Transactions on Computers, vol. 65, no. 9, pp. 2739–2751, 2016.

[9] L. He, H. Zhu, and S. A. Jarvis, “Developing graph-based co-scheduling algorithms on multicore computers,” IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 6, pp. 1617–1632, 2016.

[10] F. Bodin and S. Bihan, “Heterogeneous multicore parallel programming for graphics processing units,” Scientific Programming, vol. 17, no. 4, pp. 325–335, 2009.

[11] C.-K. Luk, S. Hong, and H. Kim, “Qilin: exploiting parallelism on heterogeneous multiprocessors with adaptive mapping,” in Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture, Micro-42, pp. 45–55, New York, NY, USA, December 2009.

[12] https://www.kernel.org/doc/Documentation/scheduler/sched-design-CFS.txt.

[13] D. Koufaty, D. Reddy, and S. Hahn, “Bias scheduling in heterogeneous multi-core architectures,” in Proceedings of the 5th ACM EuroSys Conference on Computer Systems (EuroSys ’10), pp. 125–138, ACM, Paris, France, April 2010.

[14] D. Shelepov, J. C. S. Alcaide, S. Jeffery et al., “HASS: a scheduler for heterogeneous multicore systems,” ACM SIGOPS Operating Systems Review, vol. 43, no. 2, pp. 66–75, 2009.

[15] J. Chen and L. K. John, “Efficient program scheduling for heterogeneous multi-core processors,” in Proceedings of the 46th ACM/IEEE Design Automation Conference (DAC ’09), pp. 927–930, July 2009.

[16] J. Liu, P. H. Chou, N. Bagherzadeh, and F. Kurdati, “Power-aware scheduling under timing constraints for mission-critical embedded systems,” in Proceedings of the 38th Design Automation Conference, pp. 840–845, Las Vegas, Nev, USA, June 2001.

[17] M. Ruggiero, A. Guerri, and D. Bertozzi, “Communication aware systems-on-chip allocation and scheduling framework for stream-oriented multiprocessor,” DATE pp 3–8, April 2006.

[18] J. Chen and L. K. John, “Energy-aware application scheduling on a heterogeneous multi-core system,” in Proceedings of the IEEE International Symposium on Workload Characterization (IISWC ’08), pp. 5–13, IEEE, Seattle, Wash, USA, September 2008.

[19] D. P. Gulati, C. Kim, S. Sethumadhavan, S. W. Keckler, and D. Burger, “Multitasking workload scheduling on flexible-core chip multiprocessors,” in Proceedings of the 17th International Conference on Parallel Architectures and Compilation Techniques (PACT ’08), pp. 187–196, Toronto, Canada, October 2008.

[20] M. Becchi and P. Crowley, “Dynamic thread assignment on heterogeneous multiprocessor architectures,” in Proceedings of the 3rd Conference on Computing Frontiers (CF ’06), pp. 29–39, ACM, Ischia, Italy, May 2006.

[21] D. Koufaty, D. Reddy, and S. Hahn, “Core architecture optimization for heterogeneous chip multiprocessors,” in Proceedings of the 5th European Conference on Computer Systems (EuroSys ’10), pp. 125–138, Paris, France, April 2010.

[22] H. Oh and S. Ha, “A static scheduling heuristic for heterogeneous processors,” in Euro-Par’96 Parallel Processing, vol. 1124 of Lecture Notes in Computer Science, pp. 573–577, Springer, Berlin, Germany, 1996.

[23] G. Liu, J. Park, and D. Marculescu, “Dynamic thread mapping for high-performance, power-efficient heterogeneous many-core systems,” in Proceedings of the IEEE 31st International Conference on Computer Design (ICCD ’13), pp. 54–61, October 2013.

[24] R. A. Jain and D. V. Padole, “Scalable and flexible heterogeneous multi-core system,” International Journal of Advanced Computer Science and Applications, vol. 3, no. 12, pp. 174–179, 2012.

[25] R. Kumar, D. M. Tullsen, and N. P. Jouppi, “Core architecture optimization for heterogeneous chip multiprocessors,” in Proceedings of the 15th International Conference on Parallel Architectures and Compilation Techniques (PACT ’06), pp. 23–32, ACM, Seattle, Wash, USA, September 2006.

[26] T. Constantinou, Y. Sazeides, P. Michaud, D. Fetis, and A. Seznec, “Performance implications of single thread migration on a chip multiprocessor,” ACM SIGARCH Computer Architecture News, vol. 33, no. 4, pp. 80–91, 2005.

[27] K. Van Craeynest, A. Jaleel, E. Eeckhout, P. Narvaez, and J. Emer, “Scheduling heterogeneous multi-cores through performance impact estimation (PIE),” in Proceedings of the 39th Annual International Symposium on Computer Architecture (ISCA ’12), pp. 213–224, IEEE, Portland, Ore, USA, June 2012.

[28] T. Li, D. Baumberger, D. A. Koufaty, and S. Hahn, “Efficient operating system scheduling for performance-asymmetric multi-core architectures,” in Proceedings of the ACM/IEEE Conference on Supercomputing (SC ’07), November 2007.

[29] E. K. Ardestani and J. Renau, “ESESC: A fast multicore simulator using Time-Based Sampling,” in Proceedings of the 19th IEEE International Symposium on High Performance Computer Architecture (HPCA ’13), Shenzhen, China, February 2013.

[30] C. Bienia, S. Kumar, J. P. Singh, and K. Li, “The PARSEC benchmark suite: characterization and architectural implications,” in Proceedings of the 17th International Conference on Parallel Architectures and Compilations Techniques (PACT ’08), pp. 72–81, October 2008.

[31] F. Sun, S. Ravi, A. Raghunathan, and N. K. Jha, “Synthesis of application-specific heterogeneous multiprocessor architectures using extensible processors,” in Proceedings of the 18th International Conference on VLSI Design, Kolkata, India, January 2005.
Research Article

Massive MIMO Relay Systems with Multipair Wireless Information and Power Transfer

Hongwu Liu,1 Kyung Hi Chang,2 and Kyung Sup Kwak3

1Shandong Jiaotong University, Jinan 250357, China
2Department of Electronics Engineering, Inha University, Incheon 22212, Republic of Korea
3Department of Information and Communication Engineering, Inha University, Incheon 22212, Republic of Korea

Correspondence should be addressed to Kyung Sup Kwak; kskwak@inha.ac.kr

Received 25 November 2016; Accepted 5 February 2017; Published 26 February 2017

Academic Editor: Jeongyeup Paek

Copyright © 2017 Hongwu Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper investigates destination-aided simultaneous wireless information and power transfer (SWIPT) for a decode-and-forward relay network, in which massive multiple-input multiple-output antennas are deployed at relay to assist communications among multiple source-destination pairs. During relaying, energy signals are emitted from multiple destinations when multiple sources are sending their information signals to relay. With power splitting and unlimited antennas at relay, asymptotic expression of harvested energy is derived. The analysis reveals that asymptotic harvested energy is independent of fast fading effect of wireless channels; meanwhile transmission powers of each source and destination can be scaled down inversely proportional to the number of relay antennas. To significantly reduce energy leakage interference and multipair interference, zero-forcing processing and maximum-ratio combing/maximum-ratio transmission are employed at relay. Fundamental trade-off between harvested energy and achievable sum rate is quantified. It is shown that asymptotic sum rate is neither convex nor concave with respect to power splitting and destination transmission power. Thus, a one-dimensional embedded bisection algorithm is proposed to jointly determine the optimal power splitting and destination transmission power. It shows that destination-aided SWIPT are beneficial for harvesting energy and increasing sum rate. The significant sum rate improvements of the proposed schemes are verified by numerical results.

1. Introduction

Recently, simultaneous wireless information and power transfer (SWIPT) have been envisioned for wireless relay networks [1–3]. The focuses on SWIPT for relay networks mainly include two relaying protocols, that is, time switching (TS) relaying and power splitting (PS) relaying [1]. In TS relaying, a relay harvests energy from a source-emitted RF signal and forwards source signal in a time-division manner. In PS relaying, a relay extracts energy for forwarding from its received source signal with PS operation. Generally, PS relaying can increase its spectrum efficiency by reducing the consumed time slots compared to that of TS relaying. In attempting to effectively utilize available time slots, SWIPT schemes with destination-aided energy flow (EF) have emerged [2–4]. In [2], the destination-aided EF was proposed for TS relaying protocol. In [3], an autonomous multiple-input multiple-output (MIMO) relay employing PS-based energy receiver was investigated with destination-aided EF. For cognitive relay systems, the authors in [4] proposed sending EFs from destination to energy-constrained relay.

In the above mentioned destination-aided EF transmission schemes [2, 3], only a single source-destination pair was considered. However, the demands for relay-assisted SWIPT with multiple source-destination pairs have emerged. For instance, authors in [5] proposed several power allocation schemes for a relay network, where multiple source-destination pairs communicate with each other via a common energy harvesting relay. In [6], the outage performances of SWIPT in large-scale networks with/without relay were analyzed, where multiple receivers harvest energy using PS technique. By employing multiple energy harvesting relays to assist multipair communications, the authors in [7] proposed distributed PS for SWIPT by using game theory. In [8], multiple relays were applied to assist communications from multiple base stations to multiple cell-edge users and PS-based
energy receivers were deployed at multiple relays. For wireless power transfer enabled collaborative mobile clouds, cellular data distributing among multiple mobile users was investigated in [9]. It has shown that the energy-constrained relays need to harvest energy from ambient radio-frequency (RF) signals to support multipair communications in future dense heterogeneous networks [6–10].

One challenge in relay-assisted SWIPT is a limited amount of harvested energy. Due to path-loss and inefficient RF-to-DC conversion, only a small fraction of energy emitted by a source can be harvested at relay. In order to harvest energy efficiently, smart antennas were employed in SWIPT systems (see [11] and references therein). In [12], TS and PS receivers were proposed for SWIPT in MIMO broadcasting channels. A low-complexity antenna switching for MIMO relay networks with SWIPT was investigated in [13]. Moreover, multiple antennas were deployed at relay to harvest energy from EF emitted by a single destination [3]. Although MIMO techniques have improved SWIPT performance to a certain extent, harvesting enough energy cannot be guaranteed by a limited number of antennas.

Recently, massive MIMO techniques were applied to improve wireless transmission capacity by exploiting its large array gain [14, 15]. A review of massive MIMO can be found in [16]. By employing linear signal processing [17], massive MIMO techniques have obtained increased signal-to-interference-plus-noise ratio (SINR) and power efficiency, which makes massive MIMO suitable to be deployed in practical SWIPT prototypes [18]. In [19], massive MIMO antennas were employed at a hybrid data-and-energy access point such that the minimum data rate among user nodes can be maximized whenever the number of antennas at the access point grows without bound. In [20], the authors proposed a low-complexity antenna partitioning algorithm for SWIPT systems with massive MIMO. Subjecting to a minimum harvested energy constraint, interference can be mitigated and throughput can be maximized [20]. Note that all the above mentioned works on SWIPT with massive MIMO were constrained in single-hop communications systems. To the best of our knowledge, only a few works have been conducted for multipair massive MIMO relay networks with SWIPT; for example, the authors in [21] proposed TS and PS relaying protocols for multiway amplify-and-forward (AF) relay networks, where information sharing among multiple users was aided by a massive MIMO relay.

In this paper, we propose a two-phase relaying protocol with destination-aided EFs for a decode-and-forward (DF) relay network, where multiple source-destination pairs communicate via a massive MIMO relay [10]. Different from the work of [7], where synchronization among multiple relays is required, we consider only a single relay with massive MIMO antennas and destination-aided EFs. This model has many interests, for example, in environmental sensor networks, Internet of things, and future dense heterogeneous networks, where interuser communications can be realized with the aid of an energy-constrained relay [3–10]. The main contributions of this paper can be summarized as follows:

(i) The important performance metrics including asymptotic harvested energy and asymptotic achievable rate are derived for the case in which the number of relay antennas grows without bound.

(ii) We show that destination-aided EFs are beneficial for boosting energy harvesting and, hence, the achievable sum rate can be improved significantly with the aid of destination-aided EFs. The optimal destination transmission power that maximizes the achievable sum rate is derived in closed-form. The optimal PS factor that maximizes the achievable sum rate is derived in closed-form.

(iii) We also show that by using zero-forcing (ZF) processing and maximum-ratio combining/maximal-ratio transmission (MRC/MRT) at relay, energy leakage interference (ELI) and multipair interference (MI) can be cancelled completely as the number of relay antennas grows without bound. When the number of relay antennas is large, transmission powers of each source and each destination can be scaled inversely proportional to the number of relay antennas.

(iv) We show that asymptotic sum rate is neither convex nor concave with respect to PS and destination transmission power, such that conventional convex optimization methods cannot be applied. We propose a one-dimensional embedded bisection (EB) algorithm to jointly determine the optimal PS and destination transmission power, so that the sum rate can be improved significantly.

The rest of this paper is organized as follows. Section 2 describes the system model and presents ZF processing and MRC/MRT processing. Section 3 presents the asymptotic analysis of system performance. The one-dimensional EB algorithm is also presented in Section 3. Section 4 presents numerical results and discusses the system performance of the proposed schemes. Finally, Section 5 summarizes the contributions of this study.

**Notation.** The superscripts \((\cdot)^T\), \((\cdot)^*\), and \((\cdot)^H\) represent the transpose, conjugate, and conjugate-transpose, respectively. \(E[\cdot]\) and \(\text{Var}(\cdot)\) stand for the expectation and variance operations, respectively. \(C^{MN}\) denotes the \(M \times N\) complex space and \(C.J(0, \Sigma)\) stands for the circularly symmetric complex Gaussian distribution with zero mean and variance matrix \(\Sigma\). Unless otherwise stated, vectors and matrices are, respectively, represented by bold lowercase and uppercase letters. \(1_{K}\) is the \(K\times K\) identity matrix, \(\mathbf{0}_{M \times K}\) is the \(M \times K\) zero matrix, and \(\mathbf{0}_{K}\) is the \(K \times K\) zero matrix. \(\text{Tr}(\Sigma)\) denotes trace operation of a matrix \(\Sigma\).

### 2. System Model

A block diagram of the considered multipair massive MIMO DF relay network is depicted in Figure 1, where \(K\) single-antenna equipped sources \((S_1, S_2, \ldots, S_K)\) transmit their information signals to \(K\) single-antenna equipped destinations \((D_1, D_2, \ldots, D_K)\) via an \(M\)-antenna equipped wireless-powered massive MIMO relay \((R)\). Without loss of generality, we assume that \(M\) is larger than \(K\) in this paper. We also
assume that a direct link between the source and destination of each pair does not exist due to large path-loss or obstacles. All nodes work in half-duplex mode. Moreover, noises at relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.

The relaying protocol consists of two time phases. In phase I, all source nodes transmit their information signals R for forwarding. Meanwhile, all destination nodes transmit phase I, all source nodes transmit their information signals is assumed to be small, so that it is neglected.

while noise power at relay’s energy harvesting (EH) receiver is modeled as complex zero-mean additive noises, the relay’s information detecting (ID) receiver and destinations’ receivers are modeled as complex zero-mean additive noises, while noise power at relay’s energy harvesting (EH) receiver is assumed to be small, so that it is neglected.
where $\widetilde{D}_{SR} = ((\sigma_{RP}^2 D_{SR}^{-1}/r_P P_r) + I_K)^{-1}$ and $\widetilde{D}_{DR} = ((\sigma_{RP}^2 D_{DR}^{-1}/r_P P_r) + I_K)^{-1}$. Denote the estimation error matrices of $G_{SR}$ and $G_{DR}$ by $\overline{G}_{SR}$ and $\overline{G}_{SR}$, respectively. Then, we have
\[
G_{SR} = \widetilde{G}_{SR} + \overline{G}_{SR},
\]
(9)
\[
G_{DR} = \widetilde{G}_{DR} + \overline{G}_{DR}.
\]
Since $\overline{G}_{SR}$, $\overline{G}_{SR}$, $\overline{G}_{DR}$, and $\overline{G}_{DR}$ are independent, we have $G_{SR} \sim \mathcal{CN}(0, M_{SR}K)$, $\overline{G}_{SR} \sim \mathcal{CN}(0, M_{SR}K - \widetilde{G}_{SR})$, $\overline{G}_{DR} \sim \mathcal{CN}(0, M_{DR}K)$, and $\overline{G}_{DR} \sim \mathcal{CN}(0, M_{DR}K - \widetilde{G}_{DR})$, where $\widetilde{G}_{SR}$ and $\widetilde{G}_{DR}$ are diagonal matrices whose $k$th elements are given by
\[
\sigma_{SR,k}^2 = r_P P_r \sigma_{SR,k}^2/(r_P P_r \sigma_{SR,k}^2 + \sigma_{RP}^2),
\]
and $\sigma_{DR,k}^2 = r_P P_r \sigma_{DR,k}^2/(r_P P_r \sigma_{DR,k}^2 + \sigma_{RP}^2)$, respectively. Due to wireless channel reciprocity, $G_{RD} = G_{DR}^H$ is used as the MMSE channel estimate for $G_{RD}$.

2.2. Linear Processing at Relay. The relay employs linear processing for DF relaying. In phase I, R uses a linear receiver matrix $W^H$ to separate the received signal into $K$ streams. With the linear receiver, the received signal becomes
\[
r = W^H y_{R,ID} + W^H n_R.
\]
Then, the $k$th element of $r$ is used for detecting the signal transmitted from $S_k$, which can be expressed as
\[
r_k = \sqrt{(1 - \theta) P_S W^H \overline{G}_{SR,k} x_{S_k}} + \sqrt{(1 - \theta) P_D W^H G_{DR} x_{D}} + W^H n_R.
\]
where $w_k$ and $g_{SR,k}$ are the $k$th columns of $W$ and $G_{SR}$, respectively.

After detecting the received signal, the relay node applies linear precoding to the detected signal before broadcasting. With DF relaying, the relay first decodes the original source signal and then regenerates the signal [22]. Thus, the transmitted signal at R can be expressed as
\[
x_R = \sqrt{P_R A x_S},
\]
(12)

where $A \in \mathbb{C}^{M \times K}$ is a linear precoding matrix. From (6) and (12), the received signal at $D_k$ can be written as
\[
y_{D_k} = \sqrt{P_R} G_{DR,k}^T a_k x_{S_k} + \sqrt{P_R} \sum_{j \neq k} G_{DR,k} a_j x_{S_j} + n_{D,k},
\]
(13)

where $g_{DR,k}$ and $a_k$ are the $k$th columns of $G_{DR}$ and $A$, respectively, and $n_{D,k}$ is the $k$th element of $n_D$.

2.2.1. ZF Processing. Since both the $K$ sources and $K$ destinations transmit their signals to $R$ in phase I and $R$ broadcasts the processed streams to the $K$ destinations in phase II, MI will affect both the received signals at $R$ and $K$ destinations. For the received signal at $R$, MI is the term
\[
\gamma \left( 1 - \theta \right) P_T \sum_{j \neq k} G_{SR,j}^H x_{S_j},
\]
where, for the received signal at $D_k$, MI is
\[
\sqrt{P_T} G_{SR,k}^T a_k x_{S_k}.
\]
Furthermore, the received signal at $R$ also suffers from ELI resulting from EFs transmitted from the $K$ destinations to $R$ in phase I, which can be represented by
\[
\sqrt{(1 - \theta) P_T} w_k^H G_{SR} x_{S_k}.
\]
In such a case, the relay applies the ZF receiver and ZF precoding to process the signal. By ZF processing, both MI and ELI are removed via projecting each intended information stream onto the orthogonal complement of MI and ELI. To this end, perfect CSI is required at $R$. However, when only the estimate of CSI is available at $R$, MI and ELI still exist.

The ZF processing matrices at $R$ are given by
\[
W^H = W^H_{ZF} = \left( G_{SR}^H \overline{G}_{SR} \right)^{-1} G_{SR},
\]
(14)
\[
A = A_{ZF} = \alpha_{ZF} G_{RD}^H \left( G_{RD} G_{RD}^H \right)^{-1},
\]
(15)
respectively. In (15), the normalization factor $\alpha_{ZF}$ is chosen to satisfy $|x_{D_k}|^2 = P_R$. Therefore, we have [23]
\[
\alpha_{ZF} = \sqrt{\frac{1}{\text{Tr} \left( G_{RD}^H G_{RD} \right)^{-1}}}.
\]
(16)

2.2.2. MRC/MRT Processing. Since ZF processing neglects the noise effect, the corresponding detection performance is poor when the signal-to-noise ratio (SNR) is low. By contrast, MRC/MRT processing neglects MI to maximize the received SNR. As a result, MRC/MRT processing achieves a better detection performance than that of ZF processing in the low SNR region at the cost of achieving a worse detection performance than that of ZF processing in the high SNR region. Similarly to ZF processing, MRC/MRT processes the received signal and amplifies the processed signal according to (10) and (12). The MRC receiver and MRT precoding matrices can be, respectively, expressed as [23]
\[
W^H = W^H_{MRC} = \overline{G}_{SR},
\]
(17)
\[
A = A_{MRT} = \alpha_{MRT} \overline{G}_{RD}^H.
\]
Furthermore, the normalization factor for the MRT precoding is given by [23]
\[
\alpha_{MRT} = \left( \text{Tr} \left( \overline{G}_{RD}^H \overline{G}_{RD} \right) \right)^{-1/2}.
\]
(18)

3. Asymptotic Analysis

In this section, the asymptotic harvested energy is derived for the case in which $M \to \infty$ and $K$ is fixed. The asymptotic symmetric sum rate over i.i.d. fading is derived for the case in which $M \to \infty$, $K \to \infty$, and $K/M$ is a constant.
3.1. Asymptotic Harvested Energy Analysis. To begin with, the transmission powers at each source and each destination are, respectively, scaled inversely proportional to the number of antennas at R, namely, $P_S = E_S/M$ and $P_D = E_D/M$, while keeping $E_S$ and $E_D$ fixed.

**Proposition 1.** Assume that the number of the source-destination pairs is fixed, the harvested energy at the relay as $M \to \infty$ is given by

$$E_h \xrightarrow{a.s.} \tau_0 \eta \theta (E_S \text{Tr}(D_{SR}) + F_D \text{Tr}(D_{DR})), \quad (19)$$

where $\xrightarrow{a.s.}$ denotes almost sure convergence.

**Proof.** See Appendix A.

To obtain further insights, by letting $D_{SR} = \beta_{SR} \eta \theta E_S$ and $D_{DR} = \beta_{DR} \eta \theta E_D$, the harvested energy expression in (19) over i.i.d. fading can be asymptotically expressed as

$$E_h^{\infty} = \tau_0 \eta \theta (\beta_{SR} E_S + \beta_{DR} E_D). \quad (20)$$

**Remark 2.** The asymptotic harvested energy in (20) is independent of fast fading components of wireless channels. Furthermore, the asymptotic harvested energy increases with respect to the source and destination transmission powers, respectively. Therefore, for an expected value of the harvested energy at the relay, the source transmission power can be tuned down by tuning up the destination transmission power and vice versa. Moreover, since the array gain increases monotonically with respect to $M$, both the source and destination transmission powers can be scaled down inversely proportional to $M$ without any asymptotic performance degradation.

3.2. Using a Large Receive Antenna Array. To cancel EII, orthogonal projection can be applied in detecting the desired signal. However, projecting EII into its orthogonal subspace may contaminate the desired signal. Note that if the subspace spanned by EII is orthogonal to the desired signal’s subspace when $M$ is large, the orthogonal projection scheme can work well. Thanks to the massive array implementation, the channel vectors of the desired signal and EII become nearly orthogonal when $M$ grows large. Therefore, we adopt the linear receiver as an orthogonal projection of the EII. Consequently, EII can be cancelled significantly by using the linear receiver with a large value of $M$. The main result is summarized in the following proposition.

**Proposition 3.** Assume that the number of source-destination pairs is fixed; the received signal at the relay for decoding the signal of the $k$th source as $M \to \infty$ is given by

$$y_{DR} \xrightarrow{a.s.} \sqrt{E_D} \epsilon_{DR} \eta \theta (E_S \text{Tr}(D_{SR}) + F_D \text{Tr}(D_{DR})), \quad (23)$$

for ZF processing and

$$y_{DR} \xrightarrow{a.s.} \sqrt{E_D} \epsilon_{DR} \eta \theta K^2 \beta_{SR}^2 (E_S \text{Tr}(D_{SR}) + F_D \text{Tr}(D_{DR})), \quad (24)$$

for MRC/MRT processing.

**Proof.** See Appendix B.

**Remark 4.** The result in Proposition 3 shows that ELI and MI can be cancelled completely when $M$ grows to infinity. The received signal at the relay after using linear processing includes only the desired signal. Therefore, the capacity of $S_k \to R$ link grows without bound as $M$ approaches infinity.

3.3. Using a Large Transmit Antenna Array

**Proposition 5.** Assume that the number of source-destination pairs is fixed and the transmission power at the relay is supplied by the harvested energy, the received signal at $D_k$ for decoding the relayed information signal as $M \to \infty$ is given by

$$y_{DR} \xrightarrow{a.s.} \sqrt{E_D} \epsilon_{DR} \eta \theta K^2 \beta_{SR}^2 (E_S \text{Tr}(D_{SR}) + F_D \text{Tr}(D_{DR})), \quad (23)$$

for ZF processing and

$$y_{DR} \xrightarrow{a.s.} \sqrt{E_D} \epsilon_{DR} \eta \theta K^2 \beta_{SR}^2 (E_S \text{Tr}(D_{SR}) + F_D \text{Tr}(D_{DR})), \quad (24)$$

for MRC/MRT processing.

**Proof.** See Appendix C.

To obtain further insights, the expressions in (23) and (24) of the received signal for decoding over i.i.d. fading can be, respectively, expressed as

$$y_{DR} \xrightarrow{a.s.} \sqrt{E_D} \epsilon_{DR} \eta \theta K^2 \beta_{SR}^2 (E_S \text{Tr}(D_{SR}) + F_D \text{Tr}(D_{DR})), \quad (23)$$

$$y_{DR} \xrightarrow{a.s.} \sqrt{E_D} \epsilon_{DR} \eta \theta K^2 \beta_{SR}^2 (E_S \text{Tr}(D_{SR}) + F_D \text{Tr}(D_{DR})), \quad (24)$$

where $\sigma_{DR}^2 = \epsilon_p P_f \beta_{DR}^2 / (\epsilon_p P_f \beta_{DR}^2 + \alpha_{DR}^2)$.

**Remark 6.** By using linear processing at R, the received signal at the destination $D_k$ includes only the desired signal. Therefore, the capacity of $R \to D_k$ link grows without bound, which is similar to that of $S_k \rightarrow R$ link. Furthermore, we can see that the noise and MI disappeared when $M$ grows to infinity. It can be clearly seen from (25) that the EFs emitted by the destinations provide gain for decoding the information signal. Moreover, the received desired signals in (25) are independent of small-scale fading of wireless channels.

3.4. Sum Rate and Harvested Energy Trade-Off. The end-to-end (e2e) achievable rate of the transmission link $S_k \rightarrow R \rightarrow D_k$ is given by

$$R_k = \min \{R_{SR,k}, R_{RD,k}\}, \quad (26)$$

**Proof.** See Appendix D.
where $R_{SR,k}$ and $R_{RD,k}$ are the achievable rates of the transmission links $S_k \to R$ and $R \to D_k$, respectively. From (11), the received signal used for detecting $x_{S,k}$ at the relay can be expressed as

$$r_k = \sqrt{(1-\theta)} P_S \mathbb{E} \left[ g_{SR,k}^H g_{SR,k} \right] x_{S,k} + \tilde{n}_{R,k},$$

(27)

where the effective noise $\tilde{n}_{R,k}$ is given by

$$\tilde{n}_{R,k} = \sqrt{(1-\theta)} P_S \mathbb{E} \left[ g_{SR,k}^H g_{SR,k} \right] x_{S,k} + \sqrt{(1-\theta)} P_S \sum_{j \neq k} g_{SR,j}^H g_{SR,k} x_{S,j} + \sqrt{(1-\theta)} P_D g_{DR,k}^H g_{DR,k} x_D + g_{SR,k}^H n_R.$$

(28)

It can be seen that the “desired signal” and “effective noise” in (27) are uncorrelated. Based on the fact that the worst-case uncorrelated additive noise is independent Gaussian noise of the same variance (see [17] and references therein), the achievable rate can be obtained as

$$R_{SR,k} = \log_2 \left( 1 + \frac{(1-\theta) P_S \mathbb{E} \left[ |w_{SR,k}^H|^2 \right]}{(1-\theta) P_S \mathbb{Var} \left[ w_{SR,k}^H \right] + MI_k + ELI_k + AN_k} \right),$$

(29)

$$R_{SR} = \mu K \log_2 \left( 1 + \min \left( \frac{(M-K)(1-\theta) P_S \sigma_{SR}^2}{K (1-\theta) (P_s (\beta_S - \sigma_{SR}^2) + P_D \sigma_{DR}^2), \frac{\eta \theta M (M-K) (P_S \beta_S + P_D \beta_{DR})}{\eta \theta K M \sigma_{DR}^2 (P_S \beta_S + P_D \beta_{DR}) (\beta_{DR} - \sigma_{DR}^2) + \sigma_{DR}^2 \sigma_{DR}^2} \right) \right),$$

(34)

where $\sigma_{SR}^2 = \tau_p P_p P_{SR}^2 + \sigma_{DR}^2$, $\sigma_{DR}^2 = \tau_p P_p P_{DR}^2$, and the pre-log factor $\mu = \tau_u/(2 \tau_u + \tau_p)$ resulted from the pilot transmission and half-duplex relaying.

Proof. See Appendix D.

Remark 8. We can see from (34) that the achievable rates of the dual-hop links grow without bound as $M \to \infty$.

where $MI_k$, $ELI_k$, and $AN_k$ represent the MI, ELI, and additive noise effect, respectively, and are, respectively, given by

$$MI_k \equiv (1-\theta) P_S \sum_{j \neq k} \mathbb{E} \left[ |w_{SR,k}^H|^2 \right],$$

(30)

$$ELI_k \equiv (1-\theta) P_D \mathbb{E} \left[ \|w_{SR,k}^H g_{DR,k}\|^2 \right],$$

(31)

$$AN_k \equiv \mathbb{E} \left[ \|w_{SR,k}^H n_R\|^2 \right].$$

(32)

Similarly, considering (13), the achievable rate of the $R \to D_k$ link can be expressed as

$$R_{RD,k} = \log_2 \left( 1 + \frac{\mathbb{Var} \left[ w_{DR,k}^T a_k \right]}{\sum_{j \neq k} \mathbb{Var} \left[ w_{DR,k}^T a_k \right] + (\sigma_{DR}^2)^2} \right).$$

(33)

Although (33) implies that only statistical knowledge of the channel gains has been applied at $D_k$ in decoding the information signal, the performance degradation compared to genie receiver is negligible for a large value of $M$ [17].

Theorem 7. For a finite number of relay antennas, the achievable rate of the system with ZF processing over i.i.d. fading is given by

$$R_{sum}^{co} = \mu K \log_2 \left( 1 + \min (\gamma_{SR}, \gamma_{RD}) \right),$$

(35)

Therefore, the e2e performance can be always improved by using a more large antenna array when ZF processing is employed. When $M \to \infty$, $K \to \infty$, and $K/M$ is a constant, the asymptotic sum rate over i.i.d. fading becomes

$$\gamma_{SR} \equiv \frac{(1-\theta) E_s \sigma_{SR}^2}{(1/(M/K-1)) (1-\theta) (E_s (\beta_S - \sigma_{SR}^2) + E_D \beta_{DR}) + 1/(1-K/M) \sigma_R^2},$$

(36)

$$\gamma_{RD} \equiv \frac{\eta \theta (M-K) (E_s \beta_S + E_D \beta_{DR})}{\eta \theta K \sigma_{DR}^2 (E_s \beta_S + E_D \beta_{DR}) (\beta_{DR} - \sigma_{DR}^2) + \sigma_{DR}^2 \sigma_{DR}^2}$$

(37)
denote the asymptotic SINRs of the first-hop and second-hop for each source-destination pair, respectively. This expression shows that, although the SINR of the $S_k \rightarrow R$ link is decreased by ELI, the SINR of the $R \rightarrow D_k$ link can be improved by employing destination-aided EFs. Therefore, the e2e performance can be improved when it is limited by the $R \rightarrow D_k$ link. Furthermore, we can reduce the transmission power of each source and each destination proportional to $1/M$, meanwhile maintaining a given quality-of-service.

**Proposition 9.** With ZF processing, the asymptotic sum rate is neither convex nor concave with respect to destination transmission power and the optimal destination transmission power that achieves the allowable maximum asymptotic sum rate is given by

$$E_{D}^{ZF} = \begin{cases} 0, & E_{S} \geq \frac{E_{ZF}}{\beta_{SR}} \beta_{DR} \\ \frac{E_{ZF} - E_{S}\beta_{SR}}{\beta_{DR}}, & E_{S} < \frac{E_{ZF}}{\beta_{SR}} \end{cases}$$

(38)

where $E_{ZF} \equiv (-b_1 + c_1 + \sqrt{(b_1 - c_1)^2 + 4a_1d_1})/2a_1$ with $a_1 \equiv K/(M-K)E_{SR}\sigma_{SR}^2$, $b_1 \equiv M\sigma_{SR}^2/(M-K)(1-\theta)E_{SR}\sigma_{SR}^2 - K/(M-K)$, $c_1 \equiv K\sigma_{DR}^2(\beta_{DR} - \sigma_{DR}^2)/(M-K)$, and $d_1 \equiv \sigma_{DR}^2\sigma_{DR}^2/\eta(M-K)$.

**Proof.** See Appendix E.

**Proposition 10.** With ZF processing, the asymptotic sum rate is concave with respect to $\theta$ and the optimal $\theta$ that achieves the allowable maximum asymptotic sum rate is given by

$$\theta^{ZF} = \frac{a_2 + b_2 - c_2 + d_2 - \sqrt{(a_2 + b_2 - c_2)^2 - 2(a_2 - b_2 - c_2)d_1 + d_2^2}}{2(a_2 - c_2)}$$

(39)

where $a_2 \equiv K(E_\theta(\beta_{SR} - \sigma_{SR}^2) + E_{DR}\beta_{DR})/(M-K)E_{SR}\sigma_{SR}^2$, $b_2 \equiv M\sigma_{SR}^2/(M-K)E_{SR}\sigma_{SR}^2$, $c_2 \equiv K\sigma_{DR}^2(\beta_{DR} - \sigma_{DR}^2)/(M-K)$, and $d_2 \equiv \sigma_{DR}^2\sigma_{DR}^2/\eta(M-K)(E_{SR}\beta_{SR} + E_{DR}\beta_{DR})$.

**Proof.** See Appendix F.

**Theorem 11.** With ZF processing, the asymptotic sum rate is neither convex nor concave with respect to $(\theta, E_{D})$, and there is a unique optimal $(\theta^*, E_{D}^*)$ that achieves the allowable maximum asymptotic sum rate.

**Proof.** See Appendix G.

Since the asymptotic sum rate is neither convex nor concave with respect to $(\theta, E_{D})$, the optimal $(\theta^*, E_{D}^*)$ cannot be obtained by applying conventional convex optimization methods [24]. However, Theorem 11 shows that the optimal $(\theta^*, E_{D}^*)$ is unique and Proposition 10 shows that the asymptotic sum rate is concave with respect to $\theta$, whereas Proposition 9 shows that for any given $\theta$ the corresponding optimal $E_{D}^*$ is uniquely determined. Thus, the two-dimensional searching for the optimal $(\theta^*, E_{D}^*)$ can proceed with a one-dimensional concave optimizing for $\theta^*$ with the corresponding $E_{D}^*$ explicitly determined by (38). Since bisection is effective in searching optimal solution of a concave function [24], this paper proposes a one-dimensional EB algorithm to search the optimal $(\theta^*, E_{D}^*)$ with noting that the optimal $E_{D}^*$ is embedded in one-dimensional bisection. The proposed EB algorithm is presented in Algorithm I.

To quantitatively characterize the trade-off between the harvested energy and achievable sum rate, we solve for $\theta$ in (35) and substitute it into (20). The trade-off between the asymptotic harvested energy and sum rate with ZF processing can be derived as

$$E_{h}^{ZF} = \tau_{w}\eta K(\beta_{SR}E_{S} + \beta_{RD}E_{D})$$

(40)

$$= \frac{d_2\Lambda}{c_2\Lambda + b_2\Lambda} + \frac{\Lambda}{a_2\Lambda - 1}, \quad \theta < \theta^{ZF}$$

$$= \frac{\Lambda}{a_2\Lambda - 1}, \quad \theta \geq \theta^{ZF},$$

where $a_2, b_2, c_2$, and $d_2$ are the same as those of Proposition 10 and

$$\Lambda \triangleq 2^{R_{sum}^{ZF}/\mu K} - 1.$$  

(41)

**Theorem 12.** For a finite number of relay antennas, the achievable rate of the system with MRC/MRT processing over i.i.d. fading is given by

$$R_{sum}^{MRC/MRT} = \mu K \log_2 \left( 1 + \min \left( \frac{M^2 \eta \theta \sigma_{DR}^2 (P_{SR}^2 + P_{DR}^2)}{K M \mu \theta \sigma_{DR}^2 (P_{SR}^2 + P_{DR}^2) + \sigma_{DR}^2}, \frac{M \eta \theta \sigma_{DR}^2 (P_{SR}^2 + P_{DR}^2) + \sigma_{DR}^2}{K M \mu \theta \sigma_{DR}^2 (P_{SR}^2 + P_{DR}^2) + \sigma_{DR}^2} \right) \right).$$

(42)

Thus, the e2e performance can be always improved by using a more large antenna array when MRC/MRT processing is employed. When $M \to \infty$, $K \to \infty$, and $K/M$ is constant, the asymptotic symmetric sum rate over i.i.d. fading is given by

$$R_{sum}^{MRC/MRT} = \mu K \log_2 \left( 1 + \min \left( \frac{(1 - \theta) E_{SR}\sigma_{SR}^2}{(1 - \theta) (E_{SR}^2 + E_{PR}^2)} K/M + \frac{M \eta \theta \sigma_{DR}^2 (E_{SR}^2 + E_{PR}^2)}{K M \mu \theta \sigma_{DR}^2 (E_{SR}^2 + E_{PR}^2) + \sigma_{DR}^2} \right) \right).$$

(43)

**Remark 13.** We can see from (42) that, as $M \to \infty$, the achievable rates of the dual-hop links grow without bound.
Similarly to the case of ZF processing, the above expression implies that the e2e performance with MRC/MRT processing can be improved by destination-aided EFs when it is limited by the $R \rightarrow D_k$ link.

**Proposition 14.** With MRC/MRT processing, the asymptotic sum rate is neither convex nor concave with respect to destination transmission power and the optimal destination transmission power that achieves the allowable maximum asymptotic sum rate is given by

$$E_D^{MRC/MRT} = \begin{cases} 
0, & F_S \geq \frac{E_{MRC/MRT}}{E_{SR}}, \\
E_{MRC/MRT} - E_3 \beta_{SR}, & F_S < \frac{E_{MRC/MRT}}{E_{SR}},
\end{cases} \quad (44)$$

where $E_{MRC/MRT} = (-b_3 + c_3 + \sqrt{(b_3 - c_3)^2 + 4d_3d_3})/2d_3$, $a_3 \triangleq K/M E_{SR}^2$, $b_3 \triangleq a_3^2/(1 - E_{SR})^2$, $c_3 \triangleq K \beta_{DR}/M \sigma_{DR}^2$, and $d_3 \triangleq \sigma_{DR}^2/M \eta \sigma_{DR}^2$.

**Proof.** By following similar procedure as the proof of Proposition 9 in Appendix E, Proposition 14 can be reached.

**Proposition 15.** With MRC/MRT processing, the asymptotic sum rate is concave with respect to $\theta$ and the optimal $\theta$ that achieves the allowable maximum asymptotic sum rate is given by

$$\theta_{MRC/MRT} = a_4 + b_4 - c_4 - \sqrt{(a_4 + b_4 - c_4)^2 - 2(a_4 - b_4 - c_4)d_4 + d_4^2} \quad (45)$$

where $a_4 \triangleq K (E_{SR} + E_{DR})/M E_{SR}^2$, $b_4 \triangleq a_3^2/E_{SR}^2$, $c_4 \triangleq K \beta_{DR}/M \sigma_{DR}^2$, and $d_4 \triangleq \sigma_{DR}^2/M \eta \sigma_{DR}^2$.

**Proof.** By following similar procedure as the proof of Proposition 10 in Appendix F, Proposition 15 can be reached.

**Theorem 16.** With MRC/MRT processing, the asymptotic sum rate is neither convex nor concave with respect to $(\theta, E_D)$ and there is a unique optimal $(\theta^*, E_D^*)$ that achieves the allowable maximum asymptotic sum rate.

**Proof.** By following similar procedure as the proof of Theorem 11 in Appendix G, Theorem 16 can be reached.

Since Propositions 14 and 15, and Theorem 16, have the similar expressions as those of Propositions 9 and 10, and Theorem 11, respectively, the one-dimensional EB algorithm can be applied to the case of MRC/MRT processing straightforwardly. Similarly to (40), the asymptotic harvested energy versus sum rate trade-off with MRC/MRT processing can be derived as

$$E_h^\alpha = \tau \eta K (\beta_{SR} E_S + \beta_{DR} E_D) \times \begin{cases} 
\frac{d_4 \Lambda}{c_4 \Lambda + 1}, & \theta < \theta_{MRC/MRT}, \\
1 + \frac{b_4 \Lambda}{a_4 \Lambda - 1}, & \theta \geq \theta_{MRC/MRT},
\end{cases} \quad (46)$$

where $a_4, b_4, c_4,$ and $d_4$ are the same as those of Proposition 15.

**4. Numerical Results**

This section presents some numerical results to verify the performance of the proposed protocol. For simplicity of illustration, the circuit power consumption and EH receiver sensitivity at the relay are ignored [21]. In all the illustrative examples, we choose $K = 10$, $\tau_p = 2K$, $\tau_u = 98$, $\eta = 0.7$, $p_p = 30$ dBm, and $\sigma_{R_P}^2 = \sigma_{R}^2 = \sigma_{D}^2 = -90$ dBm. Furthermore, we set $D_{SR} = D_{DR} = 70 \text{m}$ to model i.i.d. fading, where $\beta = -65$ dB corresponds to a path-loss related to a distance of about 10 m for the carrier frequency of 915 MHz.

In Figure 2, the trade-off between the harvested energy and achievable sum rate is investigated for ZF processing. Five levels of destination-aided EFs, that is, $E_{SR} = 35$ dBm, $E_{D} = 30$ dBm, $E_{D} = 20$ dBm, $E_{D} = 0$ mW, and $E_{ZF}^\alpha$ are considered, respectively, whereas $E_S$ is fixed at 25 dBm. Note that $E_D = 0$ mW corresponds to the scenario where the destinations do not transmit. The asymptotic trade-off curves are plotted by using (40). Furthermore, a set of trade-off curves are plotted by using Monte Carlo simulations with
gradually increasing $M$ from 40 to 10000. In Figure 2, the points of the trade-off curves corresponding to a higher harvested energy refer to a larger PS ratios and, consequently, this scenario leads to a higher sum rate. Nevertheless, as $\theta \to 1$, the achievable sum rates of all the fixed $E_D$ drop down dramatically and approach zero. It can be seen from Figure 2 that the trade-off between the harvested energy and sum rate can be improved significantly by using a very large antenna array. For example, for $E_D$ equal to 20 dBm and 30 dBm, destination-aided EFs can effectively improve the trade-off between harvested energy and sum rate. However, for $E_D$ = 35 dBm, destination-aided EFs deteriorate the trade-off. Furthermore, since a fixed $E_D$ can only achieve a part of the sum rate compared to that of $E^{ZF}_D$, Figure 2 shows that the curves are overlapped partially, whereas $E^{ZF}_D$ achieves the highest trade-off between harvested energy and sum rate.

In Figure 3, we investigate the trade-off between the harvested energy and achievable sum rate for MRC/MRT processing. Four levels of destination-aided EFs, that is, $E_D = 30$ dBm, $E_D = 20$ dBm, $E_D = 0$ mW, and $E^{ZF}_D$, are considered, respectively, whereas $E_h$ is fixed at 25 dBm. The asymptotic trade-off curves are plotted by using (46). The trade-off curves for $M \to \infty$ are also presented in Figure 3. As can be seen from Figure 3, the trade-off curves for $M \to \infty$ serve as the trade-off bounds. When $M$ is a limited number, similar to the case of ZF processing, a larger PS ratio leads to a higher harvested energy as well as a higher achievable sum rate. Also, the achievable sum rate drops down dramatically as $\theta \to 1$.

In Figures 4 and 5, we investigate sum rate versus $E_D$ for ZF processing and MRC/MRT processing, respectively. The curves in Figures 4 and 5 verify the correctness of Propositions 9, 10, 14, and 15. Compared with fixed $\theta$ and fixed $E_D$, the corresponding $E^{ZF}_D$ ($E^{MRC/MRT}_D$) and $\theta^{ZF}$ ($\theta^{MRC/MRT}$) can, respectively, improve the sum rate. Furthermore, Figures 4 and 5 show that the proposed one-dimensional EB algorithm achieves the allowable maximum sum rate. Figures 4 and 5 also show that the sum rate increases as $M$ increases from 40 to 10000.

In Figures 6 and 7, we investigate sum rate versus PS factor for ZF processing and MRC/MRT processing, respectively.
The results in Figures 6 and 7 verify the correctness of the analytical $\theta^{ZF}$ and $\theta^{MRC/MRT}$, respectively. As can be seen from Figures 6 and 7, the optimal PS factor almost keeps unchanged with the variable $M$ for the considered same power budget. Furthermore, the optimal PS factor for ZF processing is almost the same as that for MRC/MRT processing. Moreover, Figures 6 and 7 verify that the sum rate is concave with respect to $\theta$. Figures 6 and 7 also verify that the one-dimensional EB algorithm achieves the allowable maximum sum rate.

In Figures 8 and 9, average sum rates are plotted against source transmission power for ZF processing and MRC/MRT processing, respectively. The analytical curves are plotted by using (34) and (42) for ZF processing and MRC/MRT processing, respectively, whereas the exact sum rate curves are plotted by using Monte Carlo simulations. Figures 8 and 9 clearly show that the closed-form achievable rates in (34) and (42) are accurate in computing the sum rate based on channel statistics-aided decoding with a finite number of relay antennas. Furthermore, it can be clearly seen from Figures 8 and 9 that destination-aided EBs can improve the sum rate significantly. This verifies that ELI can be effectively cancelled out by massive MIMO processing. Moreover, the highest sum rate is achieved by the proposed one-dimensional EB algorithm. Note that $E_D^{ZF}$ and $E_D^{ZF}$ achieve the second highest sum rate in Figures 8 and 9, respectively, the correctness of Propositions 9 and 14 is verified, respectively. Also, the piecewise sum rate curves verify the correctness of Propositions 10 and 15. By comparing the curves of $M = 40$ and $M = 10000$, we observe that the gains of sum rate can be obtained by increasing the number of relay antennas while keeping the source transmission power in useful regions. Notably, Figures 8 and 9 show that ZF processing is higher than that of MRC/MRT processing in the high $E_S$ region, whereas both ZF processing and MRC/MRT processing achieve almost the same sum rate in the low $E_S$ region.

The sum rate versus the number of the relay antennas is investigated in Figure 10 with $E_S = 25$ dBm. As $M$ increases, the corresponding sum rate increases accordingly. The curves in Figure 10 show that ZF processing achieves a higher sum rate than that of MRC/MRT processing with the considered power budget. Furthermore, in the whole region of the value of $M$, the highest sum rate is achieved by the one-dimensional EB algorithm and the second highest sum rate is achieved by...
$M = 400$

$M = 10000$

$\theta = 0.3$, $E_D = 25 \text{ dBm}$, anal.

$\theta = 0.3$, $E_D = 0 \text{ mW}$, anal.

$\theta = 0.3$, $E_D = 25 \text{ dBm}$, simul.

$\theta = 0.3$, $E_D = 0 \text{ mW}$, simul.

$\theta = 0.5$, $E_D = 25 \text{ dBm}$, anal.

$\theta = 0.5$, $E_D = 0 \text{ mW}$, anal.

$\theta = 0.5$, $E_D = 25 \text{ dBm}$, simul.

$\theta = 0.5$, $E_D = 0 \text{ mW}$, simul.

$\theta = 0.5$, $E_D = 25 \text{ dBm}$, simul.

$\theta = 0.5$, $E_D = 0 \text{ mW}$, simul.

$\theta = 0.5$, $E_D = 25 \text{ dBm}$, simul.

$\theta = 0.5$, $E_D = 0 \text{ mW}$, simul.

$\theta = 0.5$, $E_D = 25 \text{ dBm}$, simul.

$\theta = 0.5$, $E_D = 0 \text{ mW}$, simul.

Figure 8: Sum rate versus $E_S$ with ZF processing.

Figure 9: Sum rate versus $E_S$ with MRC/MRT processing.

Figure 10: Sum rate versus the number of the relay antennas.

5. Conclusion

A destination-aided SWIPT relaying protocol has been proposed for a multipair massive MIMO relay network, in which a PS relay employs linear processing to cancel MI and ELI. The expressions of asymptotic harvested energy and symmetric sum rate with massive MIMO relay have been derived in closed-form. The trade-off between asymptotic harvested energy and achievable sum rate has been quantified. The effect of destination-aided EFs on the sum rate has been investigated and our results reveal that destination-aided EFs can boost the energy harvesting, so that achievable sum rate can be significantly improved by destination-aided EFs. Meanwhile, the detrimental impact of ELI on sum rate performance can be cancelled out by using a massive MIMO relay with linear processing. It has shown that asymptotic sum rate is neither convex nor concave with respect to PS and destination transmission power and a one-dimensional EB algorithm has been proposed to obtain the optimal PS and destination transmission power. The significant sum rate improvement of the proposed scheme has been verified by numerical results.

Appendix

A. Proof of Proposition 1

First, we recall two important identities for random matrices. As $M \to \infty$, the column vectors of $H_{SR}$ and $H_{DR}$ become orthogonal to each other; that is,

\[ h_{SR,i}^H h_{SR,j} \xrightarrow{M} \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \]
where $h_{SR,k}$ and $h_{DR,k}$ are the $k$th columns of $H_{SR}$ and $H_{SR}$, respectively. Consequently, as $M \to \infty$, it can be shown that

$$
\begin{array}{c}
\frac{G_{SR}^H G_{SR}}{M} \xrightarrow{a.s.} D_{SR}, \\
\frac{G_{DR}^H G_{DR}}{M} \xrightarrow{a.s.} D_{DR}.
\end{array}
$$

(A.2)

By substituting $P_S = E_S/M$ and $P_D = E_D/M$ into (4), the harvested energy at the EH receiver of R can be rewritten as

$$
E_h = \tau_d \Psi \left( E_S \text{Tr} \left( \frac{G_{SR}^H G_{SR}}{M} \right) + E_D \text{Tr} \left( \frac{G_{DR}^H G_{DR}}{M} \right) \right). \tag{A.3}
$$

Based on the fact of (A.2), whenever the number of antennas at the relay grows without bound, the harvested energy expression in (A.3) can be derived as

$$
E_h \xrightarrow{a.s.} \tau_d \Psi \left( E_S \text{Tr} \left( D_{SR} \right) + E_D \text{Tr} \left( D_{DR} \right) \right). \tag{A.4}
$$

B. Proof of Proposition 3

(1) For ZF Processing. By substituting (9) and (14) into (10), the processed signal at the relay before amplification can be expressed as

$$
\sqrt{(1-\theta)} P_S W^H G_{SR} x_S = \sqrt{(1-\theta)} P_S W^H (\hat{G}_{SR} + \delta_{SR}) x_S, \tag{B.1}
$$

By using the law of large numbers, we have

$$
\sqrt{(1-\theta)} P_S W^H \delta_{SR} x_S = \sqrt{(1-\theta)} P_S \left( \frac{G_{SR}^H G_{SR}}{M} \right)^{-1} \frac{G_{SR}^H \delta_{SR}}{M} x_S \xrightarrow{a.s.} 0, \tag{B.2}
$$

as $M \to \infty$.

Therefore, as $M \to \infty$, the received and processed signal at the relay for detection satisfies

$$
\sqrt{(1-\theta)} P_S W^H G_{SR} x_S \xrightarrow{a.s.} \sqrt{(1-\theta)} P_S x_S. \tag{B.3}
$$

The above expression can be rewritten in the element form as

$$
\sqrt{(1-\theta)} P_S W^H \bar{g}_{SR,k} x_{S,k} \xrightarrow{a.s.} \sqrt{(1-\theta)} P_S x_{S,k}, \tag{B.4}
$$

$$
\sqrt{(1-\theta)} P_S W^H \bar{g}_{SR,j} x_{S,j} \xrightarrow{a.s.} 0, \tag{B.5}
$$

for $j \neq k$.

Next, we consider the ELI term. Note that the vector $\bar{g}_{SR,k}$ is independent of the vector $\bar{g}_{DR,k}$. With ZF processing, it can be shown by using the law of large numbers that

$$
\sqrt{(1-\theta)} P_D W^H G_{DR} x_D = \alpha_{ZF} \sqrt{(1-\theta)} P_D \left( \frac{G_{SR}^H G_{SR}}{M} \right)^{-1} \frac{G_{DR}^H G_{DR}}{M} x_D \xrightarrow{a.s.} 0, \text{ as } M \to \infty. \tag{B.5}
$$

Similarly, it can be shown that

$$
W^H \mathbf{n}_R \xrightarrow{a.s.} 0. \tag{B.6}
$$

By substituting (B.4), the element form of (B.5), and the element form of (B.6) into (11), we arrive at (21).

(2) For MRC/MRT Processing. By using the law of large numbers, the desired signal in (11) converges to a deterministic value when the receive antenna count goes to infinity; that is, as $M \to \infty$,

$$
\sqrt{(1-\theta)} P_S \frac{g_{SR}^H}{M} \bar{g}_{SR,i} x_{S,i} \xrightarrow{a.s.} \sqrt{(1-\theta)} P_S g_{SR,i}^2 x_{S,i}. \tag{B.7}
$$

For the MI, as $M \to \infty$,

$$
\sqrt{(1-\theta)} P_D \frac{g_{SR}^H}{M} \bar{g}_{SR,i} x_{S,i} \xrightarrow{a.s.} 0, \forall j \neq k. \tag{B.8}
$$

We next consider the ELI, $\sqrt{(1-\theta)} P_D \frac{g_{SR}^H}{M} \bar{g}_{DR,i} x_D$. As $M \to \infty$, by using the law of large numbers, the $k$th element of the $1 \times K$ row vector, $(\sqrt{(1-\theta)} P_D/M)^{g_{SR}^H_{DR,k}}$, can be shown to be $\xrightarrow{a.s.} 0$ due to the fact that $\bar{g}_{SR,k}$ and $\bar{g}_{DR,k}$ are statistically independent. Thus, the ELI converges to 0 as $M \to \infty$; that is,

$$
\sqrt{(1-\theta)} P_D \frac{g_{SR}^H}{M} \bar{g}_{DR,i} x_D \xrightarrow{a.s.} 0. \tag{B.9}
$$

Since $\bar{g}_{SR,k}$ and $\mathbf{n}_R$ are also statistically independent, as $M \to \infty$, it can be shown that

$$
\frac{1}{M} \bar{g}_{SR,i} \mathbf{n}_R \xrightarrow{a.s.} 0. \tag{B.10}
$$

Substituting (B.7), (B.8), (B.9), and (B.10) into (11), we obtain (22).

C. Proof of Proposition 5

(1) For ZF Processing. By using [25, Lemma 2.10] and the law of large numbers, as $M \to \infty$, we have

$$
\alpha_{ZF} \xrightarrow{a.s.} \frac{M - K}{\text{Tr}(D_{RD}^{-1})}. \tag{C.1}
$$
Substituting (C.1) into (15), it can be shown that
\[
\mathbf{b}_D^{T} \mathbf{a}_k x_{SA} \xrightarrow{a.s.} M \sqrt{M - K} \frac{\text{Tr} \left( \frac{\mathbf{G}^H_R \mathbf{G}^H_{RD}}{M} \right) \sigma^2_{DR} x_{SA}}{\text{Tr} \left( \frac{\mathbf{G}^H_{RD}}{M} \right)} \tag{C.2}
\]
\[
= M \sqrt{(M - K) \text{Tr} \left( \frac{\mathbf{D}^H_{RD}}{M} \right) \sigma^2_{DR} x_{SA}}.
\]

Furthermore, by substituting (20) into \(P_R = E_h/T\), as \(M \to \infty\), we have
\[
P_R \xrightarrow{a.s.} \frac{\alpha}{\sqrt{M}} \text{Tr} (\mathbf{D}_{SR}) \xrightarrow{a.s.} \frac{\alpha}{\sqrt{M}} \text{Tr} (\mathbf{D}_{DR}) \tag{C.3}
\]
Thus, by substituting (C.2) and (C.3) into the term \(\frac{\sqrt{P_R} \mathbf{b}_D^{T} \mathbf{a}_k x_{SA}}{\sqrt{M}}\), as \(M \to \infty\), we obtain
\[
\frac{\sqrt{P_R} \mathbf{b}_D^{T} \mathbf{a}_k x_{SA}}{\sqrt{M}} \xrightarrow{a.s.} \frac{\sqrt{\alpha} \sigma_{DR}^2 \text{Tr} (\mathbf{D}^H_{RD}) (E_h \text{Tr} (\mathbf{D}_{SR}) + E_d \text{Tr} (\mathbf{D}_{DR})) x_{SA}}{\sqrt{M}} \xrightarrow{a.s.} 0, \quad \forall j \neq k, \quad \alpha \xrightarrow{a.s.} \frac{1}{\sqrt{M}} \text{Tr} (\mathbf{G}^H_R \mathbf{G}^H_{RD}/M) \xrightarrow{a.s.} 0.
\]
By substituting (C.4) and (C.5) into (13), we obtain (23).

(2) For MRC/MRT Processing. With MRC/MRT processing, by using the law of large numbers, we have
\[
\frac{1}{\sqrt{M}} \mathbf{b}_D^{T} \mathbf{a}_j x_{SA} \xrightarrow{a.s.} \sigma_{DR}^2 x_{SA} \tag{C.6}
\]
The normalization factor can be rewritten as
\[
\alpha = \frac{1}{\sqrt{M} \text{Tr} (\mathbf{G}^H_R \mathbf{G}^H_{RD}/M)} \tag{C.7}
\]
As \(M \to \infty\), it can be shown that \(\alpha \sqrt{M} \xrightarrow{a.s.} 1/\sqrt{\text{Tr} (\mathbf{D}_{DR})}\). Thus, by substituting (C.6), (C.7), and (C.3) into the term \(\alpha \sqrt{P_R} \mathbf{b}_D^{T} \mathbf{a}_j x_{SA}\), as \(M \to \infty\), we obtain
\[
\alpha \sqrt{P_R} \mathbf{b}_D^{T} \mathbf{a}_j x_{SA} \xrightarrow{a.s.} \frac{\sqrt{\alpha} \sigma_{DR}^2 \text{Tr} (\mathbf{D}^H_{RD}) (E_h \text{Tr} (\mathbf{D}_{SR}) + E_d \text{Tr} (\mathbf{D}_{DR})) x_{SA}}{\sqrt{M}} \xrightarrow{a.s.} 0, \quad \forall j \neq k, \quad \alpha \xrightarrow{a.s.} \frac{1}{\sqrt{M}} \text{Tr} (\mathbf{G}^H_R \mathbf{G}^H_{RD}/M) \xrightarrow{a.s.} 0.
\]

By substituting (C.8) and (C.9) into (13), we obtain (24).

D. Proof of Theorem 7

(i) Derive \(R_{SR,k}\). To derive analytical expression for (29), we first compute \(E\{w_k^H \mathbf{g}_{SR,k}\}, \text{Var}(w_k^H \mathbf{g}_{SR,k}), M_{\text{I}}, E_L k,\) and \(\text{AN}_{\text{R}}\), respectively.

(ii) Compute \(E\{w_k^H \mathbf{g}_{SR,k}\}\). Substituting (14) and (9) into \(w_k^H \mathbf{g}_{SR}\), we have
\[
w_k^H \mathbf{g}_{SR} = w_k^H (\mathbf{g}_{SR} + \mathbf{g}_{SR}^*) = \mathbf{I}_M + w_k^H \mathbf{g}_{SR}, \tag{D.1}
\]
which shows that
\[
w_k^H \mathbf{g}_{SR,k} = 1 + w_k^H \mathbf{g}_{SR,k}, \tag{D.2}
\]
where \(\mathbf{g}_{SR,k}\) is the \(k\)th column of \(\mathbf{g}_{SR}\). Since the zero-mean-valued \(\mathbf{g}_{SR,k}\) is independent of \(w_k\), we have \(E\{w_k^H \mathbf{g}_{SR,k}\} = 0\). Thus,
\[
E \{w_k^H \mathbf{g}_{SR,k}\} = 1. \tag{D.3}
\]

(iii) Compute \(\text{Var}(w_k^H \mathbf{g}_{SR,k})\). From (D.2) and (D.3), the variance of \(w_k^H \mathbf{g}_{SR,k}\) can be expressed as
\[
\text{Var}(w_k^H \mathbf{g}_{SR,k}) = E \left\{ \| w_k^H \mathbf{g}_{SR,k} \|^2 \right\} = (\beta_{SR,k} - \sigma_{SR,k}^2) E \left\{ \| w_k \|^2 \right\} \tag{D.4}
\]
where \(X \sim \mathcal{W}(\mathbf{G}_{SR}^H \mathbf{G}_{SR})^2\) is a \(K \times K\) central Wishart matrix with \(M\) degrees of freedom and covariance matrix \(\mathbf{I}_K\), \(\{\cdot\}_{kk}\) denotes the \((k,k)\)th element of a matrix, and the last equality is obtained by using [25, Lemma 2.10].


\[
E \left\{ \| w_k^H \mathbf{g}_{SR,j} \|^2 \right\} = E \left\{ w_k^H \mathbf{g}_{SR,j}^* \right\} = (\beta_{SR,j} - \sigma_{SR,j}^2) E \left\{ w_k^H \right\} \tag{D.5}
\]

Thus, we arrive at
\[
\text{MI}_k = \left(1 - \theta \right) P_k \sum_{j \neq k} \frac{\beta_{SR,j} - \sigma_{SR,j}^2}{(M - K) \sigma_{SR,k}^2} \tag{D.6}
\]
(iv) Compute $E_{LI_k}$. From (31), the ELI associated with the ZF receiver can be expressed as

$$E_{LI_k} = (1 - \theta) P_D E \left\{ \mathbf{w}_k^H \mathbf{G}_{DR} \mathbf{C}_{DR} \mathbf{w}_k \right\}$$

$$= (1 - \theta) P_D E \left\{ \mathbf{w}_k^H \mathbf{D}_{DR} \mathbf{w}_k \right\}$$

$$= (1 - \theta) P_D E \left\{ \|\mathbf{w}_k\|^2 \right\} \text{Tr} \left\{ \mathbf{D}_{DR} \right\}$$

$$= \frac{(1 - \theta) P_D}{(M - K)} \sum_{j=1}^{K} \beta_{DR,j}.$$

Over i.i.d. fading, the achievable rate of the transmission link $R \rightarrow D_k$ can be expressed as

$$R_{RDA} = \log_2 \left( 1 + \frac{(M - K) (1 - \theta) P_S \sigma_{SR_k}^2}{(1 - \theta) \sum_{j=1}^{K} \left( P_S (\beta_{SR_j} - \sigma_{SR_j}^2) + P_D \beta_{DR,j} \right) + \sigma_R^2} \right).$$

Substituting (D.3), (D.4), (D.6), (D.7), and (D.8) into (29), we obtain

$$R_{SRK} = \log_2 \left( 1 + \frac{(M - K) (1 - \theta) P_S \sigma_{SR_k}^2}{(1 - \theta) \sum_{j=1}^{K} \left( P_S (\beta_{SR_j} - \sigma_{SR_j}^2) + P_D \beta_{DR,j} \right) + \sigma_R^2} \right).$$

(2) Derive $R_{DRK}$. From (33), $E$ $\left\{ \mathbf{g}_{DRA}^T \mathbf{a}_k \right\}$, $\text{Var} \left( \mathbf{g}_{DRA}^T \mathbf{a}_k \right)$, and $E \left\{ \| \mathbf{g}_{DRA} \|^2 \right\}$ are needed in computing $R_{RDA}$. By applying the procedures similarly to those in deriving $R_{SRK}$, we obtain

$$E \left\{ \mathbf{g}_{DRA} \right\} = \alpha_{ZF}$$

$$\text{Var} \left( \mathbf{g}_{DRA} \right) = \frac{\alpha_{ZF}^2 (\beta_{DRA} - \sigma_{DRA}^2)}{\sigma_{DRA}^2 (M - K)}$$

$$E \left\{ \| \mathbf{g}_{DRA} \|^2 \right\} = \frac{\alpha_{ZF}^2 (\beta_{DRA} - \sigma_{DRA}^2)}{\sigma_{DRA}^2 (M - K)} \sum_{j=1}^{K} \sigma_{DR,j}^{-1} + \sigma_D^2.$$

Substituting (D.11) into (33), we obtain

$$R_{RDA} = \log_2 \left( 1 + \frac{\alpha_{ZF}^2 P_R}{\alpha_{ZF}^2 P_R (\beta_{DRA} - \sigma_{DRA}^2) / (M - K) \sum_{j=1}^{K} \sigma_{DR,j}^{-1} + \sigma_D^2} \right).$$

Over i.i.d. fading, the achievable rate of the transmission link $R \rightarrow D_k$ can be expressed as

$$R_{RDA} = \log_2 \left( 1 + \frac{(M - K) \alpha_{ZF}^2 P_R}{\alpha_{ZF}^2 P_R (\beta_{DRA} - \sigma_{DRA}^2) / (M - K) \sum_{j=1}^{K} \sigma_{DR,j}^{-1} + \sigma_D^2} \right).$$

where the asymptotic expressions for $\alpha_{ZF}$ and $P_R$ over i.i.d. fading are, respectively, given by

$$\alpha_{ZF} \xrightarrow{\text{a.s.}} \frac{M - K}{K \sigma_{RD}^2}$$

and $P_R \xrightarrow{\text{a.s.}} \frac{\eta \theta K (E_S \beta_{SR} + E_D \beta_{DR})}{\sigma_{RD}^2}$.

Then, substituting (D.10) and (D.13) into $R_{sum} = \sum_{k=1}^{K} \min \{ R_{SRK}, R_{RDA,K} \}$, we arrive at (34).

E. Proof of Proposition 9

The SINRs $\gamma_{SR}$ and $\gamma_{RD}$ in (35) can be rewritten as

$$\gamma_{SR} = \frac{1}{a_1 E_{ZF} + b_1},$$

$$\gamma_{RD} = \frac{E_{ZF}}{c_1 E_{ZF} + d_1},$$

respectively, where $E_{ZF} \equiv E_S \beta_{SR} + E_D \beta_{DR}$, $a_1 \equiv K (M - K) E_S \sigma_{SR}^2$, $b_1 \equiv M_0 \sigma_{SR}^2 (M - K) (1 - \theta) E_S \sigma_{SR}^2$, $c_1 \equiv K \sigma_{SR}^2 (\beta_{SR} - \sigma_{SR}^2)/(M - K)$, and $d_1 \equiv \sigma_{SR}^2 \sigma_{DR}^{-2} / \eta \theta (M - K)$.

From the point of view of practice, we assume that the CSI estimation error is small enough so that $\beta_{DR} > \sigma_{DR}^2$ and $c_1 > 0$ hold true. Since $E_{ZF}$ is a linear transformation of $E_D$, we first search the optimal $E_{ZF}$ instead of the optimal $E_D$. Note that $\gamma_{SR}$ and $\gamma_{RD}$ are decreasing and increasing functions with respect to $E_D \in (0, \infty)$, respectively; we consider two cases of $E_D \in (0, \infty)$. The optimal $E_{ZF}$ that achieves the allowable maximum asymptotic sum rate as well as $\gamma$ is obtained by the allowable maximum $E_{ZF}$, that is, by setting $E_D = 0$.

For the case of $E_D \in (0, \infty)$, we have $E_{SR} \geq E_{ZF} / \beta_{SR}$ by reducing $E_{SR} \in E_D <= E_{RD} \in E_D$ in such a case. Have $\gamma = \gamma_{SR} \gamma_{RD}$. For the case of $E_D \in (0, \infty)$, we have $E_{SR} > E_{ZF} / \beta_{SR}$ by reducing $E_{SR} \in E_D <= E_{RD} \in E_D$. In such a case, $\gamma_{SR} \geq \gamma_{RD}$ for $E_D \in E_{RD} \in E_D$, and $\gamma_{SR} < \gamma_{RD}$ for $E_D \in \left( E_{ZE}, \infty \right)$, where $E_{ZE}$ is the root of $\gamma_{SR} - \gamma_{RD} = 0$ with respect to $E_D$. Thus, the allowable maximum asymptotic sum rate as well as $\gamma$ is obtained by $E_{ZE}$, which can be expressed as

$$E_{ZE} = \frac{E_{ZF} - E_{PF} / \beta_{PF}}{\beta_{DR}}$$

with $E_{ZF} \equiv (-b_1 + c_1 + \sqrt{(b_1 - c_1)^2 + 4a_1 d_1})/2a_1$ by solving $\gamma_{SR} - \gamma_{RD} = 0$. This proves (38).
Since the logarithm term $\log_2(1 + y)$ inherits the convexity/concavity of $y$, determining the convexity/concavity of (35) with respect to $E_D$ is equivalent to determining the convexity/concavity of $\gamma = \min(y_{SR}, y_{RD})$ with respect to $E_D$. For the case of $y_{SR}, y_{RD} > 0$, it can be shown that $y_{RD}$ is concave with respect to $E_D \in (0, E_D^{ZF})$ and $y_{SR}$ is convex with respect to $E_D \in (E_D^{ZF}, \infty)$. Thus, $y = \min(y_{SR}, y_{RD})$ as well as the asymptotic sum rate is neither convex nor concave for such a case.

**F. Proof of Proposition 10**

The SINRs $y_{SR}$ and $y_{RD}$ in (35) can be rewritten as

\[
\begin{align*}
y_{SR} &= \frac{1 - \theta}{a_2(1 - \theta) + b_2}, \\
y_{RD} &= \frac{\theta}{c_2 \theta + d_2},
\end{align*}
\]

respectively, where $a_2 \triangleq K(E_\beta(\beta_{SR} - \sigma^2_{SR}) + E_\beta(\beta_{DR})/(M - K)E_\sigma^2_{SR}, b_2 \triangleq M\sigma^2_{SR}/(M - K)E_\sigma^2_{SR}, c_2 \triangleq K\sigma^2_{SR}(\beta_{DR} - \sigma^2_{DR})/(M - K)$, and $d_2 \triangleq \sigma^2_{DR}\eta/(M - K)(E_\beta\beta_{SR} + E_\beta\beta_{DR})$. From the point of view of practice, we assume that the CSI estimation error is small enough so that $\beta_{SR} > \sigma^2_{SR}, \beta_{DR} > \sigma^2_{DR}, a_2 > 0$, and $c_2 > 0$ hold true. Since the allowable maximum asymptotic sum rate is achieved with the allowable maximum $\gamma = \min(y_{SR}, y_{RD})$, we continue the proof by searching $\theta$ that maximizes $\gamma$.

It can be shown that $y_{SR}$ and $y_{RD}$ are decreasing and increasing functions with respect to $\theta$, respectively. Furthermore, we have $y_{SR}|_{\theta=0} > y_{RD}|_{\theta=0}$ and $y_{SR}|_{\theta=1} < y_{RD}|_{\theta=1}$. Thus, there is a single cross-point between $y_{SR}$ and $y_{RD}$ for $\theta \in (0, 1)$. Denote the cross-point by $\gamma^Z\theta$; it can be shown that $\gamma = y_{RD}$ for $\theta \in (0, \gamma^Z\theta)$ and $\gamma = y_{SR}$ for $\theta \in (\gamma^Z\theta, 1)$. Since $y_{SR}$ and $y_{RD}$ are concave with respect to $\theta \in (0, 1)$, it can be concluded that $\gamma$ is concave with respect to $\theta$. Furthermore, noting the monotonicities of $y_{SR}$ and $y_{RD}$, the allowable maximum $\gamma$ is achieved by the root of $y_{SR} - y_{RD} = 0$, which can be expressed as (39). This proves Proposition 10.

**G. Proof of Theorem 11**

We first prove that the asymptotic sum rate is neither convex nor concave with respect to $(\theta, E_D)$. Since logarithm does not change convexity/concavity, determining the convexity/concavity of (35) with respect to $(\theta, E_D)$ is equivalent to determining the convexity/concavity of $\gamma = \min(y_{SR}, y_{RD})$ with respect to $(\theta, E_D)$. It can be shown that $\gamma = y_{SR}$ exists with a nonzero probability, so that we choose to investigate the convexity/concavity of $y_{SR}$ without loss of generality. The expression of $y_{SR}$ in (36) can be rewritten as

\[
y_{SR} = \frac{1 - \theta}{a(1 - \theta)E_D + b},
\]

where $a \triangleq K_E\beta_{RD}/(M - K)E_\sigma^2_{SR}$ and $b \triangleq (KE_\beta(\beta_{SR} - \sigma^2_{SR}) + M\sigma^2_{SR}/(M - K)E_\sigma^2_{SR}$. From the point of view of practice, we assume that the CSI estimation error is small enough so that $\beta_{SR} > \sigma^2_{SR}$ and $b > 0$ hold true. Then, the Hessian of $y_{SR}$ with respect to $(\theta, E_D)$ can be expressed as

\[
\tilde{H} = \frac{2abE_D}{(b + a(1 - \theta)E_D)^3} \left[ \begin{array}{cc} 2ab(1 - \theta) & 2ab(1 - \theta) \\
2ab(1 - \theta) & 2a(1 - \theta)^3 \end{array} \right].
\]

Since $\partial^2\gamma_{SR}/\partial\theta^2 = -2abE_D/(b + a(1 - \theta)E_D)^3 < 0$ and $|\tilde{H}| < 0$, $\tilde{H}$ is neither positive definite nor negative definite. Therefore, $y_{SR}$ is neither convex nor concave with respect to $(\theta, E_D)$. For the proof of the uniqueness of the optimal $(\theta, E_D)$, we proceed by contradiction. Denote $(\theta^*, E_D^*)$ as an optimal point that achieves the maximum asymptotic sum rate. From Proposition 9, we know that $E_D^* = E_D^{ZF}(\theta^*)$ is uniquely determined so that

\[
R_{sum}(\theta^*, E_D^*) = R_{sum}(\theta^*, E_D^{ZF}(\theta^*)).
\]

From Proposition 10, we know that $\theta^* = \theta^Z(\theta^*)$ is uniquely determined so that

\[
R_{sum}(\theta^*, E_D^*) = R_{sum}(\theta^Z(\theta^*), E_D^*)).
\]

Suppose that there is another optimal point $(\tilde{\theta}^*, E_D^*)$ that achieves the maximum asymptotic sum rate with $\tilde{\theta}^* \neq \theta^*$ and $E_D^* \neq E_D^*$. From Proposition 9, we have

\[
R_{sum}(\tilde{\theta}^*, E_D^{ZF}(\tilde{\theta}^*)) < R_{sum}(\theta^*, E_D^{ZF}(\theta^*)).\]

Since $R_{sum}$ is concave with respect to $\theta$, for $\theta^*$ that achieves the maximum $R_{sum}$ with any given destination transmission power, we have

\[
R_{sum}(\tilde{\theta}^*, E_D^{ZF}(\tilde{\theta}^*)) < R_{sum}(\theta^*, E_D^{ZF}(\tilde{\theta}^*)) < R_{sum}(\theta^*, E_D^{ZF}(\theta^*)) \quad \text{and} \quad \tilde{\theta}^* \neq \theta^*.
\]

With (G.5) and (G.6), we have

\[
R_{sum}(\tilde{\theta}^*, E_D^*) = R_{sum}(\tilde{\theta}^*, E_D^{ZF}(\tilde{\theta}^*)) < R_{sum}(\theta^*, E_D^{ZF}(\tilde{\theta}^*)) < R_{sum}(\theta^*, E_D^{ZF}(\theta^*)) = R_{sum}(\theta^*, E_D^*),
\]

which contradicts the fact that $(\tilde{\theta}^*, E_D^*)$ is an optimal point.

**H. Proof of Theorem 12**

First, we derive $R_{SRA}$. Since $\mathbb{E}^{H}_{\mathbb{S}_{SRA}^H\mathbb{S}_{SRA}} = \mathbb{E}_{\mathbb{S}_{SRA}^H}\mathbb{S}_{SRA}^H\mathbb{S}_{SRA}$, it can be shown that

\[
\mathbb{E}_{\mathbb{S}_{SRA}^H\mathbb{S}_{SRA}} = \mathbb{E}_{\mathbb{S}_{SRA}^H}\mathbb{S}_{SRA}^H\mathbb{S}_{SRA} = \sigma_{SRA}^2 M.
\]

With the help of [25, Lemma 2.9], the variance of $\mathbb{E}^{H}_{\mathbb{S}_{SRA}^H\mathbb{S}_{SRA}}$ can be expressed as

\[
\text{Var}(\mathbb{E}^{H}_{\mathbb{S}_{SRA}^H\mathbb{S}_{SRA}}) = \sigma_{SRA}^2\beta_{SRA}M.
\]
Similarly, we have
\[
MI_k = (1 - \theta) P_s \sigma_{SR,k}^2 M \sum_{j=1}^{K} \beta_{SR,j},
\]
\[
ELI_k = (1 - \theta) P_D \sigma_{SR,k}^2 M \sum_{j=1}^{K} \beta_{RD,j},
\]
\[
AN_k = \sigma_R^2 \sigma_{SR,k}^2 M.
\]

By substituting (H.1), (H.2), and (H.3) into (29), we can obtain
\[
R_{SR,k} = \log_2 \left( 1 + \frac{(1 - \theta) P_s \sigma_{SR,k}^2 M}{(1 - \theta) \sum_{j=1}^{K} (P_s \beta_{SR,j} + P_D \beta_{RD,j}) + \sigma_R^2} \right). \tag{H.4}
\]

Over i.i.d. fading, the achievable rate of the transmission link \( S_k \rightarrow R \) can be expressed as
\[
R_{SR,k} = \log_2 \left( 1 + \frac{(1 - \theta) P_s \sigma_{SR,k}^2 M}{K (1 - \theta) (P_s \beta_{SR} + P_D \beta_{RD}) + \sigma_R^2} \right), \tag{H.5}
\]

where \( \sigma_{SR,k}^2 \approx \tau_p P_s \sigma_{SR,k}^2 (r_p - \beta_{SR} + \sigma_{SR,k}^2) \). Similarly, we arrive at (42).

**Competing Interests**

The authors declare that there are no competing interests regarding the publication of this paper.

**Acknowledgments**

This work was supported in part by SRF for ROCS, SEM, Shandong Provincial Natural Science Foundation, China, under Grant 2014ZR019XM, in part by the Ministry of Science, ICT and Future Planning (MSIP), Korea, under the Information Technology Research Center Support Program supervised by the Institute for Information and Communications Technology Promotion under Grant IITP-2016-H8501-16-1019, and in part by the National Research Foundation of Korea, Grant Funded by the Korean Government, MSIP, under Grant NRF-2014K1A3A1A20034987.

**References**

[1] A. A. Nasir, X. Zhou, S. Durrani, and R. A. Kennedy, “Relaying protocols for wireless energy harvesting and information processing,” IEEE Transactions on Wireless Communications, vol. 12, no. 7, pp. 3622–3636, 2013.

[2] H. Ju and R. Zhang, “User cooperation in wireless powered communication networks,” in Proceedings of the IEEE Global Communications Conference (GLOBECOM ’14), pp. 1430–1435, IEEE, Austin, Tex, USA, December 2014.

[3] Y. Huang and B. Clerckx, “Joint wireless information and power transfer for an autonomous multiple antenna relay system,” IEEE Communications Letters, vol. 19, no. 7, pp. 1113–1116, 2015.

[4] R. Sun, Y. Wang, Z. Miao, and X. Wang, “Destination-aided wireless power transfer in energy-limited cognitive relay systems,” IEEE Access, vol. 4, pp. 5385–5398, 2016.

[5] Z. Ding, S. M. Perlaza, I. Esnaola, and H. V. Poor, “Power allocation strategies in energy harvesting wireless cooperative networks,” IEEE Transactions on Wireless Communications, vol. 13, no. 2, pp. 846–860, 2014.

[6] I. Krikidis, “Simultaneous information and energy transfer in large-scale networks with/without relaying,” IEEE Transactions on Communications, vol. 62, no. 3, pp. 900–912, 2014.

[7] H. Chen, Y. Li, Y. Jiang, Y. Ma, and B. Vucetic, “Distributed power splitting for SWIPT in relay interference channels using game theory,” IEEE Transactions on Wireless Communications, vol. 14, no. 1, pp. 410–420, 2015.

[8] A. A. Nasir, D. T. Ngo, X. Zhou, R. A. Kennedy, and S. Durrani, “Joint resource optimization for multicell networks with wireless energy harvesting relays,” IEEE Transactions on Vehicular Technology, vol. 65, no. 8, pp. 6168–6183, 2016.

[9] Z. Chang, J. Gong, Y. Li et al., “Energy efficient resource allocation for wireless power transfer enabled collaborative mobile clouds,” IEEE Journal on Selected Areas in Communications, vol. 34, no. 12, pp. 3438–3450, 2016.

[10] H. Liu and K. S. Kwak, “Multipair massive MIMO relay with simultaneous wireless information and power transfer,” in Proceedings of the International Conference on Information and Communication Technology Convergence (ICTC ’16), pp. 225–230, Jeju Island, Korea, October 2016.

[11] Z. Ding, C. Zhong, D. W. K. Ng et al., “Application of smart antenna technologies in simultaneous wireless information and power transfer,” IEEE Communications Magazine, vol. 53, no. 4, pp. 86–93, 2015.

[12] R. Zhang and C. K. Ho, “MIMO broadcasting for simultaneous wireless information and power transfer,” IEEE Transactions on Wireless Communications, vol. 12, no. 5, pp. 1989–2001, 2013.

[13] I. Krikidis, S. Sasaki, S. Timotheou, and Z. Ding, “A low complexity antenna switching for joint wireless information and energy transfer in MIMO relay channels,” IEEE Transactions on Communications, vol. 62, no. 5, pp. 1577–1587, 2014.

[14] T. L. Marzetta, “Noncooperative cellular wireless with unlimited numbers of base station antennas,” IEEE Transactions on Wireless Communications, vol. 9, no. 11, pp. 3590–3600, 2010.

[15] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, “Massive MIMO for next generation wireless systems,” IEEE Communications Magazine, vol. 52, no. 2, pp. 186–195, 2014.

[16] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, “An overview of massive MIMO: benefits and challenges,” IEEE Journal on Selected Topics in Signal Processing, vol. 8, no. 5, pp. 742–758, 2014.

[17] H. Q. Ngo, H. A. Suraweera, M. Matthaiou, and E. G. Larsson, “Multipair massive MIMO full-duplex relaying with MRC/MRT processing,” in Proceedings of the 1st IEEE International Conference on Communications (ICC ’14), pp. 4807–4813, IEEE, Sydney, Australia, June 2014.

[18] X. Chen, Z. Zhang, H.-H. Chen, and H. Zhang, “Enhancing wireless information and power transfer by exploiting multi-antenna techniques,” IEEE Communications Magazine, vol. 53, no. 4, pp. 133–141, 2015.
[19] G. Yang, C. K. Ho, R. Zhang, and Y. L. Guan, “Throughput optimization for massive MIMO systems powered by wireless energy transfer,” *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 8, pp. 1640–1650, 2015.

[20] H. Wang, W. Wang, X. Chen, and Z. Zhang, “Wireless information and energy transfer in interference aware massive MIMO systems,” in *Proceedings of the IEEE Global Communications Conference (GLOBECOM ’14)*, pp. 2556–2561, IEEE, Austin, Tex, USA, December 2014.

[21] G. Amarasuriya and H. V. Poor, “Wireless information and power transfer in multi-way relay networks with massive MIMO,” in *Proceedings of the 58th IEEE Global Communications Conference (GLOBECOM ’15)*, pp. 1–7, San Diego, Calif, USA, December 2015.

[22] T. Wang, A. Cano, G. B. Giannakis, and J. N. Laneman, “High-performance cooperative demodulation with decode-and-forward relays,” *IEEE Transactions on Communications*, vol. 55, no. 7, pp. 1427–1438, 2007.

[23] H. Yang and T. L. Marzetta, “Performance of conjugate and zero-forcing beamforming in large-scale antenna systems,” *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 2, pp. 172–179, 2013.

[24] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

[25] A. M. Tulino and S. Verdù, *Random Matrix Theory and Wireless Communications: Foundations and Trends in Communications and Information Theory*, Now Publishers Inc, Hanover, Mass, USA, 2004.
Research Article

A Novel Low-Cost Real-Time Power Measurement Platform for LoWPAN IoT Devices

Yang Liu,1,2 Yubing Wang,2 Weiwei Gao,2 Wuxiong Zhang,1,2 Hua Qian,3 and Yang Yang1,2

1Shanghai Institute of Microsystem and Information Technology, Key Laboratory of Wireless Sensor Network and Communication, Chinese Academy of Sciences (CAS), Shanghai, China
2Shanghai Research Center for Wireless Communications (WiCO), Shanghai, China
3Shanghai Advanced Research Institute, Chinese Academy of Sciences (CAS), Shanghai, China

Correspondence should be addressed to Wuxiong Zhang; wuxiong.zhang@wico.sh

Received 25 November 2016; Accepted 29 January 2017; Published 22 February 2017

Copyright © 2017 Yang Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of technology and application for Internet of Things (IoT), Low-Power Wireless Personal Area Network (LoWPAN) devices are more popularly applied. Evaluation of power efficiency is important to LoWPAN applications. Conventional method to evaluate the power efficiency of different LoWPAN devices is as follows: first measure the current of the devices under working/idle/sleep state and then make an average and estimation of the lifetime of batteries, which deeply relied on the accuracy of testing equipment and is not that accurate and with high cost. In this work, a low-cost, real-time power measurement platform called PTone is proposed, which can be used to detect the real-time power of LoWPAN devices (above 99.63%) and be able to determine the state of each module of DUT system. Based on the PTone, a novel abnormal status diagnosis mechanism is proposed. The mechanism can not only judge abnormal status but also find accurate abnormal status locating and classify abnormal status accurately. According to the method, each state of Device Under Test (DUT) during wireless transmission is estimated, different abnormal status can be classified, and thus specific location of abnormal module can be found, which will significantly shorten the development process for LoWPAN devices and thus reduce costs.

1. Introduction

The rapid development of information technology, for example, 3G/4G mobile communication technology and the Internet of Things (IoT), have largely changed our everyday lives. LoWPAN devices are more popularly developed for embedded applications in biomedical electronics, wireless sensor networks, and environment monitoring [1–6]. LoWPAN devices are indispensable for modern electronic applications, and numerous hardware/software techniques have been developed for drastically reducing functional power dissipation [7–9].

In order to reduce the amount of unused power from existing devices, engineers engaged in the development of battery power-supply device are facing huge pressure to reduce the consumption of power. Many methods for reducing power consumption of the system have been proposed [10–12]. They are therefore exploring component level dynamic power consumption, which is difficult to measure, especially for low-power devices used in IoT enabled products, because of the limited dynamic range, large measurement noise, and limited bandwidth. For example, most of the battery-powered devices have low-power sleep mode that consumes very little supply current such as less than 1 μA, while the active mode usually requires more than 10 mA current. It is difficult to measure such a wide dynamic range of currents with a single measurement. Otherwise, the testing of these low-cost, low-power devices is a daunting task; since such devices are cost-sensitive, test cost is a major consideration [13, 14].

For the development of low-power devices and applications, power (or current) measurement is important. Thus, in order to fulfill the requirements for power consumption reduction, precise dynamic current waveform measurement and debug are required. In this paper, a low-cost, real-time current measurement system is proposed which is capable of digitizing a wide range of current, classifying different abnormal status, and providing specific location of abnormal
module and significantly shortening the development process. The system consists of a power measurement platform called P Tone and a novel abnormal status diagnosis mechanism which is proposed based on PTone.

A power measurement platform called P Tone can be used to detect the real-time power of LoWPAN devices and be able to determine the state of each module of DUT system which is meaningful to the R&D on LoWPAN devices/applications. By comparing the power consumption of well-performance device and DUT in corresponding state of the device system, the abnormal status can be estimated and detected. Our hardware implementation approach would be described in detail in Section 3.

In traditional abnormal status detection, the usual practice is just judging abnormal status or not. No approaches are developed to find accurate abnormal status locating and abnormal status classification for IoT devices. In order to classify different abnormal status, a novel abnormal status diagnosis mechanism based on sliding-window feature extraction (SWFE) is proposed based on P Tone.

Although much research has been done in designing systems for the detection of abnormal status to perform feature selection, to the best of our knowledge, no specific approaches have been developed for IoT devices. The work in [15, 16] developed a grained approach to perform feature selection, but enormously increasing computational complexity. A few attempts have been proposed. Wang et al. [17] developed a system to implement Bloom filters which avoids some of the computational complexity. Despite the improvement in complexity, effective Bloom filter implementation may not be possible on resource constrained embedded devices [18]. Additional approaches at improving computational complexity have been explored. McPAD [19] measures features using a sliding window to analyze pairs of bytes. References [20, 21] also use a sliding-window approach to extract features from the payload. Although these improvements obtain reliable detection results, their approaches are not able to reduce computational complexity.

First, we design a work state detection algorithm (WSDA), which is able to approximate start and end points of active time in work state by continuously moving a sliding window on the observed waveform. According to the start and end points of active time in work state, we can extract the waveform series in the work state for further analysis instead of using the whole waveform. P Tone chooses features based on unique features of each active period in work state of LoWPAN devices, which allow us to extract the features of power waveform more accurately. Then the Support Vector Machine (SVM) is used to compare the profiles of measured waveform data series for classifying different abnormal status. Finally, we investigate whether P Tone can accurately classify DUT system status based on the waveform features extracted above and verify classification accuracy through some experiments. The system considers different aspects of testing low-power device, which cannot be measured by the previous measurement due to the fact that they cannot detect the broadband low level of current waveform.

The remainder of this paper is organized as follows. Section 2 provides an overview of system description. Section 3 describes our hardware implementation approach for P Tone in detail. Section 4 describes our abnormal status classification algorithm in detail and provides evaluation of the performance of our approach. Section 5 concludes the paper.

2. System Description

2.1. Overview. Nowadays, there is strong request to design a real-time power measurement platform for LoWPAN devices, with applications to sensing in environments where it is difficult or impossible to change batteries and where the exact position of the devices might not be known.

We propose a power measurement platform called P Tone which can be used to detect the real-time power of LoWPAN devices and be able to determine the state of each module of DUT system which is meaningful to the R&D on LoWPAN devices/applications. By comparing the power consumption of well-performance device and DUT in corresponding state of the device system, the abnormal module can be estimated and detected.

P Tone consists of a low-cost power detection hardware platform and a service (such as a laptop), as shown in Figure 1.

In order to reduce the power consumption, LoWPAN devices’ wireless communication is typically divided into work state and sleep state. In work state, LoWPAN devices have several periods of active time: receiving beacon signal, sending data packet to the devices, and receiving ACK of the transmitted data packet. In sleep state, LoWPAN devices enter low-power status to extend battery life.

When a LoWPAN device receives a wireless connection request from source device, LoWPAN device takes step into work state to send ACK and then keeps quiet for a short time to wait for the preparation of source device. Work states and sleep states constitute a period of wireless communication, as shown in Figure 2. We can evaluate the status of LoWPAN devices system according to the impact of power consumption on the power waveforms change because different components in the devices have unique power consumption.
However, there are three key technical challenges. The first technical challenge is to segment the power waveforms to identify start time and end time of the active period in work state. We studied the characteristics of typical power waveforms of different kind of LoWPAN devices. We observed that the power waveforms are periodic and power consumption has obvious significant increase which causes power pulses during each active time in work state. We also observed that the period of DUT system is stable, and the intervals and amplitude of power pulses also are regular. Based on this observation, we design a power waveform feature extraction algorithm which is able to approximate start and endpoint of active time in work state by continuously moving a sliding window on the observed waveforms.

The second technical challenge is to extract the feature of power waveforms for generating classification model for each status of the DUT system. As the power consumption of components is similar, it is hard to distinguish the noise signal in the waveforms which is introduced by various factors, such as circuit noise of the DUT and measuring platform. Some conventional features such as maximum peak power, RMS deviation, the intervals between power pulses, entropy, and the number of power pulses cannot be directly used for the classification. To address this challenge, we apply a Hampel filter on the power consumption waveforms to extract the valuable signals which contain the power change caused by system perform and detect and remove the outlier signal caused by digital circuit noise. Hampel filter computes the median value of the sample and its surrounding samples. It also estimates the standard deviation of each sample about its window median using the median absolute deviation which can used to remove abnormal point in the waveform series.

The third technical challenge is to compare the profiles of filtered waveforms, which is hard to realize in real-time based on Commercial-Off-The-Shelf (COTS) power detection platform which usually just provides real-time display or records of the power consumption, so a kind of low complexity algorithm for feature comparison needs to be designed. The mean values of power in each work state align with each other because the estimated value of start and end points are different from the true value which would lead to much higher errors in mean value. Moreover, the interval time of different work state is fluctuant over time, and there may be different interval time of work state in the same system status. Therefore, mean value of power and interval time of work state are not enough to identify the system statuses. We need to combine them with more waveform features to classify system statuses.

To address this challenge, PTone uses extracted features to train a Support Vector Machine (SVM) and achieves the classification of multiple system status and uses maximum posterior probability (MAP) obtained by the SVM to estimate the system status. MAP can be used to obtain a point estimate of an unobserved quantity based on measured data.

The key novelty of this paper is on proposing the first low-cost power measure platform and an effective status detection method for LoWPAN DUT system. In this paper, we have shown that fine grained status detection method is possible by analyzing and processing measured power waveforms properly. We also have taken some experiments to prove that the techniques proposed in this paper can be used for the proposed low-cost power measure platform. Examples include the detection of two common faults of microcomputer control unit and two often encounter problems in radio frequency module in different type of LoWPAN DUTs. Our PTone techniques can be potentially used to detect a wider range of system status by building a more complete component power consumption model in the future.

### 3. Hardware Implementation

#### 3.1. Hardware Overview

The block diagram of the proposed current measurement platform is shown in Figure 3. As seen from the figure, the block diagram mainly consists of five parts:

1. Analog signal detection circuit
2. Programmable-gain instrumentation amplifier (PGA)
3. Analog-to-digital converter circuit (ADC)
4. Microcontroller unit (MCU)
5. Power-supply module

Signal detection sensor in this figure represents a sensitive resistor with very small resistance value. According to the voltage across this sensor, the power consumption of DUT can be calculated accurately. However, the instantaneous

![Figure 2: Typical working process of LoWPAN devices.](image-url)
current of LoWPAN devices is usually very small, which makes it hard to measure the current directly.

Therefore, a controllable gain amplifier is required to amplify the low voltage at the signal detection sensor output, which is represented by PGA in Figure 3. Then the analog signal amplified by PGA needs to be sampled and converts to digital signal by analog-to-digital converter circuit, which is represented by ADC in the figure.

In order to achieve wide dynamic range of gain, wide bandwidth, and low noise and high linearity and digitize a wide range of signals, it is necessary to set the value of reference voltage and input gain of PGA and output data rate of ADC to reasonable values by system control module represented by MCU in the figure. The functional characteristics of proposed system are shown in Table 1.

### 3.2 Programmable-Gain Instrumentation Amplifier (PGA)

In order to obtain suitable output voltage that can be digitized directly, PGA281 is used as the instrumentation amplifier with digital gain control in Figure 3. PGA281 offers excellent DC precision and long-term stability using modern technology with internal filters that minimize noise. The output stage connects to the low-voltage supply. The PGA is carried out by improved parallel feedback based Flipped Voltage Follower (FLV) to achieve wide dynamic range of gain, wide bandwidth, and low noise and high linearity.

The input voltage gain of PGA281 is in the range of 0.125 to 128 in binary steps. The output stage has a gain adjustment lower (FLV) to achieve wide dynamic range of gain, wide bandwidth, and low noise and high linearity and digitize a wide range of signals.

According to Nyquist Theorem, the input analog signal can be recovered without distortion; thus voltage of input analog signal \( U_{\text{ADCin}} \) can be measured based on the value of \( N_{\text{ADC}} \).

According to the Nyquist Theorem, the input analog signal can be recovered without distortion; thus voltage of input analog signal \( U_{\text{ADCin}} \) can be measured based on the value of \( N_{\text{ADC}} \).

### 3.3 Analog-to-Digital Converter (ADC)

Analog-to-digital converter circuit used ADS1259 to digitize the signals at PGA281 output. ADS1259 outputs 24 bits of conversion data in binary format. Combined with a signal amplifier, a high-resolution, high-accuracy measurement system is formed that is capable of digitizing a wide range of signals.

Depending on different input values, the ADS1259 has corresponding output code \( N_{\text{ADC}} \), a 24-bit hexadecimal number. If the voltage of input signal \( U_{\text{ADCin}} \) is larger than positive full-scale input \( V_{\text{REF}} \), the output code of ADC is set to 7FFFFFh. \( U_{\text{ADCin}} \) is provided by the PGA and equal to the output voltage of PGA; namely, \( U_{\text{PGAout}} = U_{\text{ADCin}} \). If the input voltage of ADC is less than \( -V_{\text{REF}}(2^{23}/(2^{23} - 1)) \), the output code of ADC is set to 800000H. And if the input voltage is in the interval of \( (-V_{\text{REF}}(2^{23}/(2^{23} - 1)), V_{\text{REF}}) \), the input signal can be digitized to a dynamic output code, which can be expressed as

\[
N_{\text{ADC}} = U_{\text{ADCin}} \left( \frac{2^{23} - 1}{V_{\text{REF}}} \right). \tag{4}
\]

### 3.4 Microcontroller Unit (MCU)

In order to configure the value of reference voltage and input gain of PGA and output data rate of ADC and calculate the data obtained from ADC outputs, a low-power and high-speed MCU, namely, STM32L151, is used in the system control module of the proposed system. STM32L151 device has high-performance ARM core operating at a 32 MHz frequency, as well as high-speed embedded memories having up to 128 Kbytes of flash memory and up to 16 Kbytes of RAM. Moreover, the

![Figure 3: Block diagram for implementation.](image-url)
STM32L151 device contains standard and advanced communication interfaces, such as USARTs and USBs, which can transmit measured data to computer for record and further analysis. Due to the good performance and low-cost of STM32L151 device, the proposed system is well represented in the configuration of a variety of parameters of the system and data processing.

3.5. Power-Supply Module. A power-supply module has been elaborated in order to satisfy the power requirement of PGA, ADC, and MCU.

The PGA281 requires three supply voltages: the high-voltage analog supply, the low-voltage output amplifier supplies, and the digital I/O supply. The high-voltage analog supplies, which include negative high-voltage supply (VSN) and positive high-voltage supply (VSP), power the high-voltage input stage section and are set to ±12 V in the proposed system. The voltage of USB is 5 V, which cannot provide ±12 V power supply, so that a DC–DC boosting circuit based on MC34063 is proposed. The power-supply module can provide stable power which improve the stability of the system.

3.6. Parameter Measured by the Proposed System. The proposed system is developed as a low-cost, high-resolution, high-accuracy, real-time measurement system for the power LoWPAN devices. It can measure and estimate instantaneous current, average current, average power, and instantaneous power. The instantaneous current of the DUT is measured in a sampling interval, which is given by

\[ I_{\text{DUTin}}(n) = \frac{N_{\text{ADC}}(n) \left( V_{\text{REF}} / (2^{23} - 1) \right) (R + R_{\text{PGAin}})}{A_{\text{PGA}} (R \times R_{\text{PGAin}})} \]  

\[ I_{\text{DUTin}}(n) \] stands for the current value in \( n \)th sampling interval.

The average current value \( I_{\text{average}} \) is computed once per cycle \( N_s \), which can be calculated by

\[ I_{\text{average}} = \frac{1}{N_s} \sum_{n=1}^{N_s} \left( I_{\text{DUTin}}(n) \right) = \frac{1}{N_s} \sum_{n=1}^{N_s} \left( \frac{N_{\text{ADC}}(n) \left( V_{\text{REF}} / (2^{23} - 1) \right) (R + R_{\text{PGAin}})}{A_{\text{PGA}} (R \times R_{\text{PGAin}})} \right). \]  

The input impedance of PGA is very large and the impedance of \( R \) is very small, which is usually set to 10 Ω; that is, \( R_{\text{PGAin}} \gg R \). Therefore, the instantaneous current of DUT can be simplified to the following:

\[ I_{\text{DUTin}}(n) = \frac{U_{\text{REF}}}{R} \cdot \frac{N_{\text{ADC}}(n) \left( V_{\text{REF}} / (2^{23} - 1) \right)}{A_{\text{PGA}} \times R}. \]  

Similarly, the average current value can be expressed as follows:

\[ I_{\text{average}} = \frac{1}{N_s} \sum_{n=1}^{N_s} \left( \frac{N_{\text{ADC}}(n) \left( V_{\text{REF}} / (2^{23} - 1) \right)}{A_{\text{PGA}} (R \times R_{\text{PGAin}})} \right). \]  

Assuming the battery \( E \) of DUT is ideal, and the voltage of the battery is not time-varying; that is, \( U_E \) is a constant. The average power of the DUT is calculated once per cycle \( N_s \), which is shown as follows:

\[ P_{\text{average}} = I_{\text{average}} U_E - \left( I_{\text{average}} \right)^2 \frac{R \times R_{\text{PGAin}}}{R + R_{\text{PGAin}}} \]  

Due to the fact that \( R_{\text{PGAin}} \gg R \) and \( R \to 0 \), the equation can be expressed as

\[ P_{\text{average}} = I_{\text{average}} U_E = U_E \times \frac{1}{N_s} \sum_{n=1}^{N_s} \left( I_{\text{DUTin}}(n) \right). \]  

The instantaneous power is the power consumption of DUT in each sampling slot, which is given by

\[ P(n) = U_E \times I_{\text{DUTin}}(n). \]  

The measured instantaneous power can be set to computer through Universal Asynchronous Receiver/Transmitter (UART) for further research.

In all equations, \( V_{\text{REF}} \) is the difference between the voltage values of positive reference absolute input and negative reference absolute input of ADC. \( N_s, R, \) and \( A_{\text{PGA}} \) are known parameters. \( N_{\text{ADC}}(n) \) is a value measured by ADC in a sample interval, and according to \( N_{\text{ADC}}(n) \), we can measure and calculate the value of the instantaneous current, average current, average power, and instantaneous power of DUT. The description of the symbols in the equations above is summarized in Table 2.

| Symbols | Meaning |
|---------|---------|
| \( U_E \) | Analog voltage of signal detection sensor |
| \( R_{\text{DUTin}} \) | Input impedance of DUT |
| \( R_{\text{PGAin}} \) | Input impedance of PGA |
| \( A_{\text{PGA}} \) | Gain of PGA |
| \( U_{\text{ADCin}} \) | Analog input voltage of ADC |
| \( U_{\text{PGAout}} \) | Analog output voltage of PGA |
| \( N_{\text{ADC}} \) | Digital output of ADC |
| \( I_{\text{DUTin}}(n) \) | Digital current value in \( n \)th sampling interval provided by MCU |
| \( V_{\text{REF}} \) | Reference voltage of PGA controlled by MCU |

As shown in Figure 4, DUT is a kind of LoWPAN device, which periodically builds a wireless connection with another LoWPAN device. The proposed system is connected to the DUT via an interface reserved for power measurement. The measured data will be calculated and transmitted to computer for record and further analysis via USB cable. The battery can supply the power of DUT.
4. Status Detection and Classification

4.1. Measurement Accuracy. We compare our measured waveforms with Keysight 34465A millimeter. Keysight 34465A millimeter has high sensitivity and accuracy, so that it can be used as the ground-truth of the measured waveforms and the qualification standard of the accuracy of our measured data. We can see that these values measured by our platform are close to 34465A millimeter. The result is shown in Figure 5(a).

We quantitatively evaluate the errors in Figure 5(b), which shows a cumulative distribution function (CDF) of the difference between PTone’s measurement and the Keysight 34465A millimeter’s measurement.

The CDF shows that the 90th percentile error is 4 mw. Our results further show that PTone’s mean power measurement error is 0.37 mw. Since the average power in our experiments is 100 mw, on average, PTone estimates power to within 0.37% of its correct value.

We confirm the performance of our system through this experiment. Besides we can ensure the evident effectiveness of the proposed system to make it easier to measure instantaneous current, average current, instantaneous power, and average power of DUT, as well as checking for system status during the period of development of LoWPAN devices.

4.2. Power Consumption Waveforms Information. LoWPAN devices contain a variety of components, such low-power MCU, RF circuits, power management module, and sensor module. In a real-world application, the sensor module often adopts COTS sensors, which have mature and stable power consumption model and have little interfere to the detection of DUT system status. MCU and RF circuits consume most of the power offered by power supply [22, 23]. Each LoWPAN DUT uses the power in maintenance of system configuration and sends ACK or data packet during the work state. After this period, the DUT system reduces the workload of MCU and shuts down RF circuits and then enters the sleep state which has almost no electric energy loss.

The PTone continually monitors the power waveforms of the DUT in real time, and the information of the waveforms can be represented by the following equation:

$$ P(i) = P_M(i) + P_R(i) + N(i) \quad i \in [0, T]. \quad (12) $$

In the equation above, $P_M(i)$ represents power consumption of MCU module and $i$ stands for the $i$th sample point. $P_R(i)$ represents power consumption of RF module. $P_M(i)$ and $P_R(i)$ make up a large proportion of the total power $P(i)$ and the noise $N(i)$ comes from random measurement noise and the power consumption of sensors and components in peripheral circuit and so on.

This section is organized as follows. First, we remove the noise in the measured waveform. Second, we detect the work state in the waveform which includes the start time and end time of active time. Third, we select and extract features based on the character of LoWPAN devices’ power consumption. Finally, we use the extracted features to train a multiple classifier, which can detect and classify the status of LoWPAN devices accurately.

4.3. Noise Removal. Even though the proportion of system noise in the total measured power waveform is not large, the noise cannot be ignored due to amplitude excitations of noise with large dynamic range, as shown in Figure 6(a). The goal of this preprocessing step is to dampen the noise signal and improve the signal-to-interference-and-noise ratio (SINR) of the $P_M(i)$ and $P_R(i)$ signal.

The frequency of system noise is not regular and occurs in a random fashion. In order to minimize the influence of system noise to the following feature extraction and classification, we use an appropriate filter to remove noise in such a situation.

If we use a Butterworth low pass filter, a type of signal processing filter widely used to remove high frequency noise, the large amplitude noise would make the power waveform distortion, because the noise signal in the waveforms is introduced by various factors, such as test environment, circuit noise of the DUT, and measuring platform, and is not all of high frequency. To address this challenge, PTone applies a Hampel filter to the measured power waveforms, to detect and remove outliers. Hampel filter computes the median value of the sample and its surrounding samples by using a sliding window. It also estimates the standard deviation of each sample about its window median using the median absolute deviation. If a sample differs from the median by more than three standard deviations, it is replaced with the media, which can eliminate the outliers caused by noise [24, 25].

The result shows that Hampel filter can remove the outliers effectively under the premise of retaining the valid data.

4.4. Work State Detection. In order to detect work state and distinguish the active time and quiet time in work state, we need to detect the start points and end points of the active time work state of DUT system. We process the measured data in a causal system, which is different from traditional mean absolute deviation (MAD) algorithm. MAD
algorithm is a kind of edge detection algorithm widely used in the digital image processing and not suitable for the time signals processing in this paper. To address this challenge, we propose a work state detection algorithm (WSDA).

First, the algorithm calculates weighted mean absolute value in the latest $W$ sample points. The weighted coefficients of each point in the window width of $W$ are set according to the distance to the current point. The weighted mean absolute value (WMAV) is calculated using the following equation:

$$m(i) = \sum_{j=0}^{W-1} \omega(j) P'(i-j), \quad i \in [0, T').$$

In the equation, $j$ represents the distance to the current point in the slipping window, $P'(i-j)$ represents the filtered data of $P$ waveforms processed by a Hampel filter, $\omega(j)$ represents the weighted coefficients of each point in the window and decreases with the increase of distance $j$, and $m(i)$ represents WMAV of the current point, which can be used to detect the start and end points of work state.

Second, after calculating the WMAV of each point in the measured power consumption waveforms, WSDA compares the WMAVs with a threshold $\theta_{th}$. Let $\Delta m(i) = m(i) - m(i-1)$ and $\delta(i) = |\Delta m(i)| - \theta_{th}$; if $\Delta m(i) > 0$ and $\delta(i) > 0$, it indicates that the WMAV has a significant increase in interval $[i-W, i]$ and current point can be regarded as a candidate of state point.

If $\Delta m(i) < 0$ and $\delta(i) > 0$, it indicates an obvious decrease in the interval, which can be regarded as a candidate of end point. If $\delta(i) \leq 0$, it shows that the variation of WMAVs
**Input:** Sequence \( P' \) of \( T' \) points  

**Output:** Start points \( s_w \) of each work state  

1. **Assert** \( s_f = 0 \)  
2. \( s_w \leftarrow \text{FINDCANDIDATES} \left( P' \right) \)  
3. **For** \( k \leftarrow 0 \) to \( N_w \) **do**  
   4. If \( \left( \frac{1}{s_w(N_w)} - s_w(k) \right) \sum_{j=s_w(k)}^{s_w(N_w)} \Delta m(j) > \theta_{th} \)  
   5. \( s_f = 1 \)  
6. return \( s_w \)  
7. procedure FINDCANDIDATES \( (P') \)  
8. **For** \( i \leftarrow 0 \) to \( T' \) **do**  
   9. \( m(i) \leftarrow \sum_{j=0}^{W-1} \omega(j)P(i-j) \)  
   10. \( \Delta m(i) \leftarrow m(i) - m(i-1) \)  
   11. \( \delta(i) \leftarrow |\Delta m(i)| - \theta_{th} \)  
   12. If \( \delta(i) > 0 \) and \( \Delta m(i) > 0 \) **do**  
   13. add \( i \)th point into start point candidate set \( \{s_w(1), s_w(2), s_w(3), \ldots, s_w(N_w)\} \) sequentially  
   14. return \( \{s_w(1), s_w(2), s_w(3), \ldots, s_w(N_w)\} \)

**Algorithm 1:** Work state detection algorithm (WSDA).

...does not exceed the threshold \( \theta_{th} \) and power consumption waveforms do not change obviously in the interval.

Third, in a real-world measurement, there may be more than one state point candidate on the rising edge of power waveforms, so we need to choose an appropriate candidate as the start point in each work state. Due to the requirement of the complete work state waveforms for feature extraction, it is inappropriate to use the mid-value of those candidates which is commonly used in statistics, and we chose the state points according to the following ways. Let \( \{s_w(1), s_w(2), s_w(3), \ldots, s_w(N_w)\} \) represent the set of start point candidates obtained from start point detection algorithm. \( s_w(1) \) represents the first start point candidate in chronological order. \( N_w \) represents number of candidates in the set.

WSDA compares the \( \Delta m(i) \) values of each point in \( \{s_w(1), s_w(2), s_w(3), \ldots, s_w(N_w)\} \) on the same rising edge. If the mean value of\( \Delta m(i) \) from point \( s_w(k) \) to point \( s_w(N_w) \) is larger than threshold \( \theta_{th} \), we define that point \( s_w(k) \) has stable growth subsequent waveforms as shown in the following equation:

\[
\frac{1}{s_w(N_w) - s_w(k)} \sum_{j=s_w(k)}^{s_w(N_w)} \Delta m(j) > \theta_{th}, \quad k \in [1, N_w].
\]

Then, we choose the minimum value in \( s_w(1), s_w(2), s_w(3), \ldots, s_w(N_w) \) which satisfies the above equation as the start point of the active period of work state and set the work state flag \( s_f = 1 \) to capture the increasing and decreasing trends. The end points \( e_w \) of each work state are obtained in a similar way. The pseudocode of our algorithm is presented in Algorithm 1.

Lastly, our algorithm detects the start point and end point candidates in the whole waveform and chose appropriate ones as start or end points to confirm the presence of a complete active time of work state within that interval, as shown in Figure 7.

The figure shows that even though the filtered waveform has some outliers, our algorithm can detect the start point accurately in different work state.

![Figure 7: Work state detection.](image)

**Figure 7:** Work state detection.

### 4.5 Feature Selection and Extraction.

In order to differentiate between statuses of the DUT system based on the real power consumption, we need to extract some unique features that can classify different system status. And according to the start and end points of active time in work state, we can extract the waveform series in the work state for further analysis instead of using the whole waveform, which allows us to extract...
the features of power waveform more accurately. Feature selection techniques are important to select features for good classification results. Previous works on feature selection were based on filter or wrapper methods [26–28]. However, the methods need to process a large number of data to find out appropriate features, which is hard to achieve in real-time and low-cost platform.

Instead, PTone chooses the features based on unique features of each active period in work state of LoWPAN devices.

As shown in Figure 8, S1, S2, S3, and S4 represent the active time of receiving beacon for source node, sending ACK, sending data, and receiving ACK from source node, respectively. Work state includes S1, S2, S3, and S4 section and the quiet time between them.

$T_w$ represents the duration time of work state. $T_s$ represents the period of DUT system which includes one work state and one sleep state. And we observed that $T_w$ is fluctuant and increases with the increase of transmission data. We also observed that $T_s$ is stable even though it is controlled by MCU of the DUT system and in real-world application, $T_s$ is far greater than $T_w$, and the change of $T_w$ has no great effect on $T_s$. In the same experimental condition, if outliers are presented within $T_s$ in each DUT period, that is, the measured value of power consumption is either too large or too small, we usually may consider that abnormal configuration of DUT’s MCU exists in the wake-up time control module or system clock of the MCU. Therefore, $T_s$ can be selected as a feature to evaluate the MCU performance of the DUT.

We also observed that the power waveforms in S2 and S3 have similar amplitude in each work state. If the DUTs do not perform well in radio frequency module, the amplitude of S2 and S3 may be different and not stable. So the difference of maximum value of S2 and S3 can be used to evaluate the status of radio frequency (module of DUT devices).

After many experiments and analysis, we carefully selected some distinctive features of LoWPAN devices to classify the system status of DUT as follows:

**Waveform Features**

- Mean amplitude of S2 and S3
- Root mean square deviation of S2 and S3
- Mean amplitude of S1 and S4
- Quiet time in work state
- System period $T_s$

PTone can detect and classify the abnormal system status of LoWPAN devices based on the features proposed above. Note that we only detect the abnormality which affects the performance of low-power wireless communication, and the abnormality is inconvenient and hard to be detected in traditional ways. The labeled features are collected in the following ways. We connect a LoWPAN DUT into our hardware platform. Then, we start a source device which could search the LoWPAN DUT and build wireless communication. Then, the LoWPAN device enters work state and sleep state periodically. Meanwhile, the real-time power waveform sequences will be measured by PTone. The work state, the start time, and end time of S1, S2, S3, and S4 and sleep state in the sequences can be detected by the work state detection algorithm (WSDA) proposed above. Then, PTone extracts the labeled features from the measured power waveform sequences based on the start and end point of each active time. Then, the features could be trained for the further analysis and classification of system status.

**4.6. Classification and Evaluation.** In this section, we investigate whether PTone can accurately classify DUT system status based on the waveform features extracted above. Based on our experience in the R&D process of LoWPAN devices, we divided system abnormal status into several groups: two kinds of RF module abnormal cases and two kinds of MCU control system abnormal cases.

RF module abnormal cases include the case of abnormal power control and the case of insufficient RF power in certain active period of work state. MCU control system abnormal cases include the case of irregular interval of each active time in work state and the case of inappropriate DUT’s system period, which is caused by crystal deviation, synchronization strategy bugs, and so on.

The existence of abnormal power control, which is represented by RF-1, undermines the network connectivity and even may lead to the failure of data transmission. A typical waveform of this status is shown in Figure 9(a). The amplitude of S2 is not smooth, which should be stable during the active
period to have better performance according to the design of the devices. The power waveform indicates that the power control in the S2 period during work state is abnormal.

The insufficient RF power, which is represented by RF-2, may reduce the coverage of wireless network and influence the use of LoWPAN devices. A typical waveform of this status is shown in Figure 9(b). Even though the amplitudes of S2 and S3 are smooth in each active time, the amplitudes have obvious difference, which should be similar during sending packet and sending ACK period according to our design. The insufficient RF power in S3 causes this kind of abnormal power waveform.

The unstable interval of active time may be caused by inaccurate configuration in MCU, incomprehensive estimation of time-delay in DUT system, and so on, which is represented by M-1. In LoWPAN, every device takes a time slot to receive beacon and send ACK or data package. Unstable interval of the active time can result in communication conflicts with other devices. A typical waveform of this status is shown in Figure 10(a).
The figure shows that the interval between S2 and S3 is much larger than that of well-performed devices. S3 is repeated two times in the same work state, which means that the excessive interval between S2 and S3 leads to unnecessary retransmission of packet data (S3). This phenomenon is difficult to be directly observed in the traditional test method.

Time synchronization is a key issue for any distributed system and is widely used in distributed wireless sensor networks, especially LoWPAN. So, if the system period is not stable and accurate, not only would power consumption be improved, but also the device would have a bad effect on the communication of other devices in the area with high node density. The case of abnormal system period is represented by M-2. A typical waveform of this status is shown in Figure 10(b). The figure shows that the system in M-2 does not have a suitable sleep state between each work state, and the device receives beacon and sends data all the time, which is not in accordance with the design of the devices.

The extracted features are used to train Support Vector Machine (SVM) and achieve the classification of multiple system status. We use maximum posterior probability (MAP) obtained by the SVM to estimate the system status. MAP can be used to obtain a point estimate of an unobserved quantity based on measured data. It is closely related to Fisher’s method of maximum likelihood (ML) estimation but employs an augmented optimization objective which incorporates a prior distribution (that quantifies the additional information available through prior knowledge of a related event) over the quantity one wants to estimate [29–32].

We visualize the output of the classification as follows: we choose two of the features and calculate the maximum posterior probability (MAP) of each status in a 2D plane. The axes are the two features or combinations of the two features.

In Figure 11(a), the MAP of M-1 and M-2 status are estimated on the basis of two features, that is, the quiet time between S2 and S3 and system period. Due to the limit of using only two features, the SVM only can classify the system status into 3 kinds of cases, M-1, M-2, and uncertain status. The uncertain status includes RF-1, RF-2, normal status, and other unknown statuses. Every point in Figure 10 has a posterior probability of each system status. And the color of point is the MAP value of a system with those two features. If the MAP of system in certain status is larger than in others, we can classify the system in that status.

For example, if a DUT’s quiet time between S2 and S3 is 60 ms, and system period is 700 ms, we can classify that the system status is in M-2 status, because the MAP of that point in M-2 status is more than 90%.

We also visualize the output of the classification by calculating the MAP based on different features to estimate system status. We classify RF-1 and RF-2 (Figure 11(b)) and M-2 and RF-1 (Figure 11(c)) by using mean value of S2, quiet time between S2 and S3, and system period.

Note that we can accurately classify two system statuses by using two only two features.

In the actual course of application, we use more than two features to train the multiple classifier, which can detect different system status at the same time.

Our method in the real-world application is composed of offline step and online step. In the offline step, we collect waveform data of DUT in known statuses, extract the features, and train classifier. In the online step, we connect the DUT (in unknown status) into our platform. Then, the real-time power waveform of the DUT can be transmitted into a server, which is a Lenovo laptop with a Core i5 desktop CPU and 4 GB RAM. Then, the server calculates the MAP of each status based on the well-trained classifier and classifies the real-time system status based on the following rules. $P_N, P_{M-1}, P_{M-2}, P_{RF-1},$ and $P_{RF-2}$ represent the MAP of normal, M-1, M-2, RF-1, and RF-2 status, respectively. If the difference between the two largest values of $P_N, P_{M-1}, P_{M-2}, P_{RF-1},$ and $P_{RF-2}$ is more than a certain threshold $\theta_{MAP}$, the system is assigned to the status with largest MAP. If there are no such values, P'Tone classifies the system into uncertain status and waits for further data to calculate the MAP of each status. The threshold $\theta_{MAP}$ reduces the probability of wrong classification, but, at the same time, it also increases the number of data collected from the DUT and the time consumption of analyzing data. According to our measurements, $\theta_{MAP}$ is set to 20% in our experiments.

To evaluate P'Tone’s system status classification accuracy, we collect 200 power waveform sequences (with the same length) from different statuses (40 sequences in each status). We train a type of system status classifier based on some features proposed and extracted above. The average accuracy of classification and some typical multiple classification of system status are shown in Figure 12.

In Figure 12(a), the figure shows that the average accuracy of multiple classification is 89.5%. The classification of abnormal system period status (M-2) has higher accuracy (95%) because the time of sleep state between each work state, as the main feature of M-2, is longer than that of work state and can be easily distinguished. Meanwhile, the classification accuracy of RF-2 is relatively low (82.5%), because the amplitude difference between S2 and S3 in work states, which is one of the main features of RF-2, is usually very small and easily affected by noise. Even though there is some noise in the measured sequences and limited performance of P'Tone’s server, our platform can still achieve an accuracy of 89.5%.

In Figure 12(b), the figure shows that the multiple classifier can calculate the MAP of each status. In the 1st measurement, the MAP of system in normal status is 85%, which is higher than that of M-1 (2%), M-2 (3%), RF-1 (3%), and RF-2 status (7%). According to the classification rule proposed above, $P_N - P_{RF-2} > \theta_{MAP}$, we can classify the DUT system in the 1st measurement into normal status. Similarly, the DUT system in the 2nd, 3rd, 4th, and 5th measurement can be classified into M-1, M-2, RF-1, and RF-2 status, respectively.

The results of the experiment show that P'Tone can detect and classify typical system at high classification rate by using different features extracted from power waveforms.
5. Conclusions

In this paper, a low-cost, instantaneous current/power waveform detection platform with desirable resolution and accuracy is proposed. Some power definitions, for example, average current, instantaneous current, average power, and instantaneous power, can be measured and calculated precisely. A novel and convenient testing method based on this system has been demonstrated for LoWPAN devices. The platform can be designed as a hermetically sealed module with economical components with small volume, which can be used in a variety of scenes whenever a wide range of small DC current is to be measured.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research is partially supported by the National Natural Science Foundation of China (NSFC) under Grants 61471346 and 61461136004, the Science and Technology Commission of Shanghai Municipality (STCSM) under Grant 14ZR1439700, the International Cooperation project under Grant 2014DFE10160, the National High Technology Research
Mobile Information Systems 13

Multiple Classification Average Accuracy

| Classification accuracy (%) | Normal | M-1 | M-2 | RF-1 | RF-2 |
|---------------------------|-------|-----|-----|------|------|
| 50                        |       |     |     |      |      |
| 55                        |       |     |     |      |      |
| 60                        |       |     |     |      |      |
| 65                        |       |     |     |      |      |
| 70                        |       |     |     |      |      |
| 75                        |       |     |     |      |      |
| 80                        |       |     |     |      |      |
| 85                        |       |     |     |      |      |
| 90                        |       |     |     |      |      |
| 95                        |       |     |     |      |      |
| 100                       |       |     |     |      |      |

Typical measurements

| Maximum posterior probability | 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
|-------------------------------|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Typical measurements          | 1 | 2   | 3   | 4   | 5   |     |     |     |     |     |   |

(a) Multiple Classification Average Accuracy

Multiple classification based on extracted features

and Development Program 863 under Grant 2015AA01A702, the Project of Shanghai Municipality under Grant 16DZ2348600, and the Shanghai Innovation Action Plan under Grants 15511102602.

References

[1] Md. Sakhawat Hossen, A. F. M. Sultanul Kabir, R. H. Khan, and A. Azfar, "Interconnection between 802.15.4 devices and IPv6: implications and existing approaches," IFCS International Journal of Computer Science Issues, vol. 7, no. 1, pp. 19–31, 2010.

[2] Z. Popovic, E. A. Falkenstein, D. Costinett, and R. Zane, “Low-power far-field wireless powering for wireless sensors,” Proceedings of the IEEE, vol. 101, no. 6, pp. 1397–1409, 2013.

[3] K. Keikhosrav, P. Kamalinejad, S. Mirabbasi, K. Takahata, and V. C. M. Leung, “An ultra-low-power monitoring system for inductively coupled biomedical implants,” in Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS ’13), pp. 2283–2286, May 2013.

[4] A. B. Dolgov and R. Zane, "Low-power wireless medical sensor platform," in Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS ’06), pp. 2067–2070, New York, NY, USA, September 2006.

[5] C. Villani, D. Balsamo, D. Brunelli, and L. Benini, “Ultra-low power sensor for autonomous non-invasive voltage measurement in IoT solutions for energy efficiency,” in Smart Sensors, Actuators, and MEMS VII; and Cyber Physical Systems, vol. 9517 of Proceedings of SPIE, pp. 93–95, May 2015.

[6] J. Wang, J. Li, S. L. Ho et al., “Study and experimental verification of a rectangular printed-circuit-board wireless transfer system for low power devices,” IEEE Transactions on Magnetics, vol. 48, no. 11, pp. 3013–3016, 2012.

[7] X. Wen, “Low-power testing for low-power devices,” in Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT ’10), October 2010.

[8] P. Girard, N. Nicolici, and X. Wen, Power-Aware Testing and Test Strategies for Low Power Devices, Springer, Boston, Mass, USA, 2010.

[9] S. Davidson, “About the power problem [review of ‘power-aware testing and test strategies for low power devices’ (Girard, P., Eds., et.; 2010)],” IEEE Design & Test of Computers, vol. 27, no. 6, pp. 72–73, 2010.

[10] Y. Higami, S. Y. Kobayashi, and Y. Takamatsu, “A method to reduce power dissipation during test for sequential circuits,” in Proceedings of the 11th Asian Test Symposium (ATS ’02), pp. 326–331, Hagåtña, Guam, USA, November 2002.

[11] N. Dharmaweera and R. Parthiban, “Reducing power consumption in an optical circuit-switched core network by switching off amplifiers,” in Proceedings of the 22nd Wireless and Optical Communications Conference (WOCC ’13), pp. 532–537, Chongqing, China, May 2013.

[12] C. Kim and S.-M. S. Kang, “A low-power reduced swing single clock flip-flop,” in Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS ’01), pp. 806–809, New South Wales, Australia, May 2001.

[13] C. P. Ravikumar, M. Hirech, and X. Wen, “Test strategies for low power devices,” in Proceedings of the Design, Automation and Test in Europe (DATE ’08), pp. 728–733, Munich, Germany, March 2008.

[14] C. Krügel, T. Toth, and E. Kirda, “Service specific anomaly detection for network intrusion detection,” in Proceedings of the 2002 ACM symposium on Applied computing (SAC ‘02), pp. 201–208, Madrid, Spain, March 2002.

[15] G. Ortiz, M. Leibl, J. Huber, and J. W. Kolar, “Design and experimental testing of a resonant DC-DC converter for solid-state transformers,” IEEE Transactions on Power Electronics, vol. PP, no. 99, p. 1, 2016.
[16] K. Wang and S. J. Stolfo, “Anomalous payload-based network intrusion detection,” Lecture Notes in Computer Science, vol. 3224, pp. 203–222, 2004.

[17] K. Wang, J. J. Parekh, and S. J. Stolfo, “Anagram: a content anomaly detector resistant to mimicry attack,” in Recent Advances in Intrusion Detection, vol. 4219 of Lecture Notes in Computer Science, pp. 226–248, Springer, Berlin, Germany, 2006.

[18] H. Chen, Y. Chen, and D. H. Summerville, “A survey on the application of FPGAs for network infrastructure security,” IEEE Communications Surveys and Tutorials, vol. 13, no. 4, pp. 541–561, 2011.

[19] R. Perdisci, D. Ariu, P. Fogla, G. Giacinto, and W. Lee, “McPAD: a multiple classifier system for accurate payload-based anomaly detection,” Computer Networks, vol. 53, no. 6, pp. 864–881, 2009.

[20] D. Ariu, R. Tronci, and G. Giacinto, “HMMPay: an intrusion detection system based on Hidden Markov Models,” Computers and Security, vol. 30, no. 4, pp. 221–241, 2011.

[21] D. H. Summerville, K. M. Zach, and Y. Chen, “Ultra-lightweight deep packet anomaly detection for Internet of Things devices,” in Proceedings of the 34th IEEE International Performance Computing and Communications Conference (IPCCC ’15), Nanjing, China, December 2015.

[22] X. Xie, Q. Zhou, K. Li, A. Beling, and J. Campbell, “1.8 watt RF power and 60% power conversion efficiency based on photodiode flip-chip-bonded on diamond,” in Proceedings of the 2014 Conference on Lasers and Electro-Optics (CLEO ’14), San Jose, Calif, USA, June 2014.

[23] F. Knutti, N. Tobler, and H. Mathis, “Low-power voting device for use in education and polls employing TI’s CC2530 RF CHIP,” in Proceedings of the 6th European Embedded Design in Education and Research Conference (EDERC ’14), pp. 221–224, Milano, Italy, September 2014.

[24] R. K. Pearson, Y. Neuvo, J. Astola, and M. Gabbouj, “The class of generalized hampel filters,” in Proceedings of the 23rd European Signal Processing Conference (EUSIPCO ’15), pp. 2501–2505, Nice, France, August 2015.

[25] J.-X. Zhang, Q.-H. Zhong, Y.-P. Dai, and Z. Liu, “A new denoising method based on wavelet transform and transforming Hampel filter,” in Proceedings of the SICE 2003 Annual Conference, vol. 2, pp. 2147–2151, Fukui, Japan, 2003, IEEE Cat. No.03TH8734.

[26] J. Li, X. Li, and W. Zhang, “A filter feature selection method based LLRFC and redundancy analysis for tumor classification using gene expression data,” in Proceedings of the 12th World Congress on Intelligent Control and Automation (WCICA ’16), pp. 2861–2867, Guilin, China, June 2016.

[27] H. Hosni and F. Mhamdi, “A filter correlation method for feature selection,” in Proceedings of the 25th International Workshop on Database and Expert Systems Applications (DEXA ’14), pp. 59–63, September 2014.

[28] N. El Aboudi and L. Benhlima, “Review on wrapper feature selection approaches,” in Proceedings of the International Conference on Engineering & MIS (ICEMIS ’16), Agadir, Morocco, September 2016.

[29] A. Bendale and T. Boult, “Reliable posterior probability estimation for streaming face recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW ’14), pp. 56–63, Columbus, Ohio, USA, June 2014.

[30] D. Wang and Y. Zhou, “Sparse posterior probability support vector machines,” in Proceedings of the IEEE Workshop on Statistical Signal Processing (SSP ’14), pp. 396–399, Gold Coast, Queensland, Australia, July 2014.

[31] C. P. Shapiro, “Classification by maximum posterior probability,” The Annals of Statistics, vol. 5, no. 1, pp. 185–190, 1977.

[32] R. Odai, Y. Taniguchi, M. Kumoi et al., “Multivalued document classification by Relevance Vector Machine in terms of maximum posterior probability,” in Abstracts of Annual Conference of Japan Society for Management Information, p. 37, The Japan Society for Management Information (JASMIN), 2010.
Research Article

Small-World and Scale-Free Network Models for IoT Systems

Insoo Sohn

Division of Electronics and Electrical Engineering, Dongguk University, 26 Pil-dong 3-ga, Chung-gu, Seoul 100-715, Republic of Korea

Correspondence should be addressed to Insoo Sohn; isohn@dongguk.edu

Received 21 November 2016; Accepted 10 January 2017; Published 30 January 2017

Academic Editor: Jeongyeup Paek

Copyright © 2017 Insoo Sohn. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

It is expected that Internet of Things (IoT) revolution will enable new solutions and business for consumers and entrepreneurs by connecting billions of physical world devices with varying capabilities. However, for successful realization of IoT, challenges such as heterogeneous connectivity, ubiquitous coverage, reduced network and device complexity, enhanced power savings, and enhanced resource management have to be solved. All these challenges are heavily impacted by the IoT network topology supported by massive number of connected devices. Small-world networks and scale-free networks are important complex network models with massive number of nodes and have been actively used to study the network topology of brain networks, social networks, and wireless networks. These models, also, have been applied to IoT networks to enhance synchronization, error tolerance, and more. However, due to interdisciplinary nature of the network science, with heavy emphasis on graph theory, it is not easy to study the various tools provided by complex network models. Therefore, in this paper, we attempt to introduce basic concepts of graph theory, including small-world networks and scale-free networks, and provide system models that can be easily implemented to be used as a powerful tool in solving various research problems related to IoT.

1. Introduction

We are experiencing explosive growth in digital data and connected devices today. The estimated number of connected devices connected to the Internet is predicted to reach approximately 6.4 billion in 2016 and 20.8 billion by the year 2020 according to Gartner. Since the inception of the term Internet of Things (IoT), which was intended originally for radio frequency identification (RFID) network, in 1999 by the Auto-ID Center of MIT [1], the concept has evolved into a network that is exponentially larger and more complex, supporting a large number of devices with services such as automotive, utilities, logistics, healthcare, and public safety using various technologies such as wireless sensor network, Wi-Fi, ZigBee, and cellular network to connect to the Internet backbone.

To realize the IoT revolution that will enable new solutions and business for consumers and entrepreneurs by connecting billions of physical world devices with varying capabilities, many major standards development organizations (SDOs) such as ITU-T, IETF, IEEE, and 3GPP have been very active on IoT standardization. In 2012, Recommendation ITU-T Y.2060, “Overview of the Internet of Things” was approved by ITU-T Study Group 13 (SG 13) that provides an IoT reference model and defines IoT as a global infrastructure for the information society, enabling advanced services by interconnecting things based on existing and evolving interoperable information and communications technologies [2]. IETF is actively working on a set of IoT standards focusing on IP transmission over NFC, Bluetooth, and WPAN [1]. IEEE P2413 WG is working on IoT architectural framework standardization that includes IoT domains, abstractions, and commonalities [1]. Finally, 3GPP is working on a narrowband evolution of LTE for IoT (NB-IoT) to be included in Rel. 13 that will further reduce cost and power but provide extended coverage compared to the LTE-M introduced in Rel. 12. Despite the different views on the IoT standardization by different SDOs, there is a general consensus on the technical challenges the IoT needs to solve. These challenges are to provide heterogeneous connectivity, ubiquitous coverage, reduced network and device complexity, enhanced power savings, and enhanced resource management. All these challenges are heavily impacted by the IoT network topology supported by massive number of connected devices.

Network science is a branch of science of interdisciplinary nature based on graph theory, statistical mechanics, data
mining, and more [3]. Network science utilizes various models and tools to study complex networks’ topology such as brain networks, social networks, and wireless networks. For decades, random network or Erdős-Rényi network has been the main network model for studying real world complex networks. A random network of \( N \) nodes is constructed by connecting \( N(N - 1)/2 \) node pairs if a randomly generated number is greater than a probability \( p \). Recently, the “small-world effect” or “six degrees of separation” principle, which was first discovered by the social psychologist Milgram in [4], was first studied by Watts and Strogatz in [5]. In small-world networks, by randomly reconnecting a small number of links in a regular lattice network, the average path length is reduced significantly [6–12]. Both the random network and small-world network have homogeneous network topology where the nodes have approximately the same number of links. In recent years, scientists have discovered that many real world networks such as world wide web (WWW), social networks, and metabolic networks are not random with node connection or edge distribution approximated by Poisson distribution but have power-law distribution [13–21]. In contrast to the Poisson distribution, the power-law distribution has higher peaks and “fat” tails describing the existence of few nodes with massive links observed in real networks. In 1999, Barabási and Albert proposed the scale-free network that has edge distribution equal to power-law nature. The two main features of the scale-free network are that it is an evolving network with incoming nodes and that these nodes are attached preferentially to the existing nodes with a large number of links.

Due to interdisciplinary nature of the network science, with heavy emphasis on graph theory, it is not easy to study the various tools provided by complex network models. There are numerous small-world and scale-free network introductory research papers from statistical mechanics branch, but there are very few introductory research papers from information and communications technology branch that can be used for wireless network optimization. Therefore, in this paper, we attempt to introduce basic concepts of graph theory, including small-world networks and scale-free networks, and provide system models with wireless channel characteristics that can be easily implemented to be used as a powerful tool in solving various research problems related to IoT. Furthermore, we evaluate the proposed system models of small-world networks and scale-free networks based on various complex network metrics, such as average path length and clustering coefficient.

The remainder of the paper is organized as follows. Section 2 presents recent research activities related to applications of small-world and scale-free concepts to IoT networks. Section 3 provides an overview on basic concepts related to complex networks. In Section 4, we describe the proposed system model for small-world networks and system model for scale-free network in Section 5. In Section 6, we present various network characteristics that are needed to be added to the conventional complex network models for IoT modeling. The numerical results are presented in Section 7. Finally, we conclude in Section 8.

2. Related Works

Small-world network and scale-free network models have been applied to various wireless networks, serving as different basis to the IoT platform, to solve various problems. We briefly describe recent research activities devoted to improving the wireless networks’ performance based on the small-world and scale-free concepts. One of the first known works that has applied the small-world concept to wireless networks research is the work done in [22] by Helmy. It was shown that the path length of wireless networks can be drastically reduced by rewiring a small number of links between the wireless nodes. Furthermore, a novel resource discovery method was proposed for small-world concept based wireless networks. Authors in [23] proposed to improve synchronization in wireless sensor networks by applying small-world concepts. The proposed model utilizes a small number of high end sensors (H-sensors) with long communication range in addition to the large number of low end sensors (L-sensors) in the network. Using a modified flooding time synchronization protocol in the proposed heterogeneous sensor network, a small-world topology is realized. Experimental results have shown that the synchronization error can be reduced by more than 50% compared to the conventional methods.

Another important application of the small-world concept is in improving the energy efficiency in wireless networks. In designing an ad hoc network with improved energy efficiency, the authors in [24] used a new energy efficiency metric of the wireless nodes for the proposed small-world network. The small-world properties were verified through simulation analysis and the proposed model was shown to be more energy efficient than the conventional random network. One of the advantages of scale-free network is the robustness against node failures. In [25], the authors proposed two protocols for constructing scale-free networks. The two protocols are called preferential attachment with time varying feedback and opportunistic two-stage random attachment. Based on degree distribution analysis, it was shown that the proposed network model is scale-free. Authors in [26] proposed a new scale-free network that includes node removal process with compensation mechanism compared to the conventional Barabási and Albert (BA) network that only considers node addition step for node evolution process. The new model is called neighborhood log-on and log-off (NLL) model due to the similar mechanism as in the action of logging on and logging off a system. It was shown that the proposed NLL model achieves decreased average path length and enhanced network connectivity compared to the BA model. In [27], a new scale-free network called flow-aware scale-free (FASF) model is proposed. In this work, the wireless sensor network is modeled as a weighted network and the traffic of the nodes is modeled as the weight. The performance of the proposed model was compared to the BA and NLL models and was shown to achieve increased network lifetime and reduced average path length. Authors in [28] proposed a new shortcut strategy based on local importance of nodes. The proposed method was evaluated in a sensor network with regular sensor nodes and super sensor nodes and was shown to have small-world features. In [29], small-world network...
models were applied to mobile ad hoc network (MANET) and were implemented based on routing protocols in OPNET. The distance vector based routing protocols and link-state based routing protocols were compared and it was shown that link-state based protocols converge faster. Authors in [30] proposed an energy aware BA model for wireless sensor networks (WSN). The key idea of the proposed method is to consider both node degree and residual energy in preferential attachment. In [31], the authors proposed two protocols to construct energy-efficient and robust large-scale WSN with scale-free properties. The first scheme modifies the BA model by integrating clustering and degree constraint. The second scheme improves energy efficiency through avoiding links to hub-nodes with large potential degrees. The performance of the proposed protocols was verified based on scale-free property, average path length, energy efficiency, and network robustness.

3. Basic Concepts

Complex networks such as computer networks, sensor networks, brain networks, and social networks can be represented as a graph. A graph consists of vertices, nodes, or points connected by edges, links, or lines. Mathematically, a graph can be represented by ordered pairs set \( G = (V, E) \), where \( V \) is a set of \( N \) vertices and \( E \) is a set of edges connecting elements of \( V \). Furthermore, the degree \( K \) of a vertex is defined as the total number of edges connected to that vertex. Figure 1 shows a graph consisting of 7 vertices and 7 edges. The degree of node 4 is equal to \( K = 3 \), since there are 3 edges connected to nodes 2, 3, and 5. The graph in Figure 1 can be described as a network with a set of vertices \( V = \{1, 2, 3, 4, 5, 6, 7\} \) and a set of edges \( E = \{(1, 2), (1, 3), (2, 4), (3, 4), (4, 5), (5, 6), (6, 7)\} \).

An important model of complex network is the random network or Erdős-Rényi network model. Random network model is commonly used as a network reference model to compare a newly proposed network model. A random network can be characterized by the total number of nodes \( N \) and the probability \( p \) that two nodes are connected. A random network can be constructed by the following procedure with \( N \) nodes and connection probability \( p \).

**Step 1.** Start with a ring of \( N \) nodes.

**Step 2.** Choose a node pair \((i, j)\).

**Step 3.** If a randomly generated number between 0 and 1 is less than \( p \), then connect the two nodes \( i \) and \( j \).

**Step 4.** Repeat Steps 2 and 3 for all possible node pairs \( N(N - 1)/2 \).

To evaluate the size of a network, the average path length is used and is defined as the average distance between two vertices, averaged over all possible pairs of vertices. The distance between a pair of vertices is defined to be the number of minimum edges or hops connecting the two vertices. For example, the distance between vertices 1 and 5 in the network in Figure 1 is equal to 3. The average path length is determined as follows:

\[
L = \left( \frac{N}{2} \right)^{-1} \sum_{i \neq j} l_{ij},
\]

where \( N \) is the total number of vertices in the network, \( l_{ij} \) is the minimum distance between vertices \( i \) and \( j \), and \( \left( \frac{N}{2} \right) \) represents all possible numbers of pairs of vertices. Figure 2 shows a ring of vertices with \( N = 4 \) and \( K = 3 \). All possible pairs of vertices for \( N = 4 \) network are \( E = \{(1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)\} \) with 6 pairs of possible edges. As shown in the figure, all the possible connections have been implemented in the network. Using (1), the average distance is calculated as \( L = 6^{-1} \times (1 + 1 + 1 + 1 + 1 + 1) = 1 \). The clustering coefficient is defined as the average fraction of pairs of neighbor vertices that are also neighbors of each other. The clustering coefficient measures the cliquishness of a typical friendship circle. The average clustering coefficient averaged over all vertices \( i = 1 \cdots N \) is given by

\[
C = \frac{1}{N} \sum_{i=1}^{N} C_i,
\]

where \( C_i \) is the clustering coefficient for vertex \( i \) defined as

\[
C_i = \frac{2E_i}{K_i(K_i - 1)},
\]

where \( E_i \) is the actual number of edges connecting the neighbors of vertex \( i \), \( K_i \) is the total number of neighbor vertices connected to vertex \( i \), and \( K_i(K_i - 1)/2 \) is the maximum number of possible connections between the neighbor vertices. Therefore, \( C_i \) represents a ratio of actual neighbor node connections to the maximum possible connections. For the network shown in Figure 2, for vertex 1, \( K_1 = 3 \) and \( E_1 = 3 \), and the maximum number of connections between the neighbor vertices is equal to \( 3(3 - 1)/2 = 3 \). Thus, \( C_1 = 3/3 = 1 \).

The degree distribution is another important metric used for network topology analysis. It is defined as the probability that a randomly chosen node \( i \) has a degree \( k \). A network’s degree distribution depends on the total number of vertices \( N \) in the network and the node connection probability \( p \). A random graph with large \( N \) and \( p \) has a degree distribution...
that can be approximated as Poisson distribution represented as follows:

\[ p(k) = e^{-\langle k \rangle} \frac{\langle k \rangle^k}{k!}, \]

where \( k \) is the degree and \( \langle k \rangle \) is the average degree. Figure 3 shows the degree distribution for a random network with \( N = 1000, p = 0.01 \), and \( \langle k \rangle = 10 \). It is seen from the figure that the shape of the distribution has a peak around \( \langle k \rangle = 10 \) and falls off exponentially to the sides. In contrast to degree distribution of random networks and small-world networks following Poisson distribution, scale-free networks have degree distribution that follows power-law distribution defined as

\[ p(k) = k^{-\gamma}, \]

where \( k \) is the degree and \( \gamma \) is the scale-free exponent. Many real networks have scale-free property with power-law distribution. For example, Internet, actor casting, and paper citation have power-law distribution with \( \gamma = 2.1, \gamma = 2.3 \), and \( \gamma = 3 \), respectively.

4. Small-World Network

In this section, we discuss the design and implementation issues related to modeling small-world networks through algorithm description, system model presentation, and metric calculation module overview.

4.1. Algorithm. The small-world model has been actively applied to the communications networks research due to resulting network topology with features such as smaller average transmission delay and more robust network connectivity. The small-world network is constructed by randomly rewiring the edges of a ring lattice with \( N \) nodes. The following procedure describes the basic steps of the small-world network construction. By varying the rewiring probability \( p \), one can analyze the transition of the network from a lattice structure to a random structure with \( 0 \leq p \leq 1 \).

**Step 1.** Start with a ring of \( N \) nodes.

**Step 2.** Connect \( K \) nearest nodes for all the nodes \( i = 1 \cdots N \).

**Step 3.** Reconnect the edges to a randomly chosen node with probability \( p \).

**Step 4.** Repeat Step 3 for all \( NK/2 \) edges in the ring network.

4.2. System Model. Figure 4 shows the system model for implementing a small-world network. The system contains five major blocks: node initialization block, node connection block, rewiring block, average path length calculation block, and clustering coefficient calculation block. Parameters \( N, K \), and \( p \) correspond to total number of nodes, initial degree of all the nodes, and rewiring probability, respectively. Furthermore, the node connection matrix or adjacency matrix gives information about all the node connections after the completion of the rewiring process. The node connection matrix shown in equation (6) describes the node connections for the graph example in Figure 1.

\[
\begin{array}{ccccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 \\
2 & 1 & 0 & 0 & 1 & 0 & 0 \\
3 & 1 & 0 & 0 & 1 & 0 & 0 \\
4 & 0 & 1 & 1 & 0 & 1 & 0 \\
5 & 0 & 0 & 0 & 1 & 0 & 1 \\
6 & 0 & 0 & 0 & 0 & 1 & 0 \\
7 & 0 & 0 & 0 & 0 & 0 & 1 \\
\end{array}
\]

Note that the small-world network is modeled by a relational graph where the distance is based on edges or hops rather than the absolute distance used in spatial graphs.

4.3. Metric Calculation Modules. In average path length calculation block, based on the node connectional matrix, which contains all the connections between the nodes, the number of hops needed to reach a node \( j \) from node \( i \)
needs to be calculated. The first step in this module is to find all possible node pair index \((i, j)\). For all the node pairs indices, we start by checking if there is a direct connection between nodes \(i\) and \(j\). If there is no direct connection, we check for 2-hop connection where node \(j\) is connected through an intermediate node. We continue this process with increasing number of hops until all the numbers of connections for all the node pair indices have been found. Finally, all the numbers of hops found for all the node pairs are added and divided by the total number of node pairs. For a network with very large number of nodes, breadth-first search (BFS) algorithm [32] is recommended for average path length calculation. The basic idea of BFS algorithm is to label a reference node as “0” and then “ripple” the labeling process until all the nodes have been labeled. The labels provide the distance with reference to node 0.

In clustering coefficient calculation block, based on the node connection matrix, the total number of neighbor nodes connected to node \(i\) is found. Using the number of neighbor nodes found, the maximum number of possible connections is calculated by \(K_i(K_i - 1)/2\). The next step is to find the actual number of edges connecting the neighbors of node \(i\). We continue this process for all \(N\) nodes. Finally, we use (3) to calculate the clustering coefficient for node \(i\) using the information found in the previous steps and get the final clustering coefficient using (2).

5. Scale-Free Network

In this section, we discuss the design and implementation issues related to modeling scale-free networks through algorithm description, system model presentation, and metric calculation module overview.

5.1. Algorithm. Major features of scale-free networks that are different compared to random networks and small-world networks are dynamic addition of new nodes and preferential attachment to existing nodes with rich connections. Due to these features, in contrast to random networks and small-world networks with Poisson distribution, scale-free networks have degree distribution following power-law nature, resulting in higher probability of finding nodes with a large number of links. The following algorithm shows the steps towards construction of a scale-free network.

Step 1. Start with a small number \(m_0\) nodes with degree \(K\).

Step 2. Introduce a new node into the network.

Step 3. Connect the new node to \(m\) existing nodes based on maximum degree probability shown as follows:

\[
\Pi(k_i) = \frac{k_i}{\sum_j k_j},
\]

where \(\Pi(k_i)\) is the probability of selecting node \(i\), \(k_i\) is the degree of node \(i\), and \(\sum_j k_j\) is the total number of edges in the current network.

Step 4. Repeat Steps 2 and 3, until a network with \(N = t + m_0\) nodes and \(mt\) edges has been constructed.

5.2. System Model. The system model for constructing a scale-free network is shown in Figure 5. The main modules are initialization modules, node connection modules, and metric calculation modules. The system model starts by initially connecting \(m_0\) nodes with \(K\) edges. The node connection matrix is generated based on the initial network topology. Based on the node connection matrix, an edge vector is created. We implement the preferential attachment step or Step 3 of the scale-free network algorithm by using the edge vector. The edge vector contains multiple node indices, where the number of index repetitions indicates the number of edges of that node. The edge vector based method is further described based on Figure 6 and equations (8), (9), (10), and (11).

(i) Initial Node Connection

**Connection Matrix**

\[
\begin{bmatrix}
0 & 1 & 1 \\
1 & 0 & 1 \\
1 & 1 & 0
\end{bmatrix}
\]  

(8)

**Edge Vector**

\[
1, 1, 2, 2, 3, 3
\]  

(9)
(ii) Network with New Node 4

Connection Matrix

\[
\begin{bmatrix}
0 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 \\
\end{bmatrix}
\] (10)

Edge Vector

\[1 \ 1 \ 2 \ 2 \ 3 \ 3 \ 4 \ 4\] (11)

In Figure 6, an initial graph is shown with 3 vertices and 2 edges for each vertex. Equations (8), (9), (10), and (11) show the connection matrix with connection information of the initial graph topology. Furthermore, an edge vector includes node indices of all the nodes in the graph in multiples of twos, since all the nodes have 2 links connected to the neighbor nodes. To implement the maximum degree based node selection with \(m = 2\) target nodes, two random numbers between 1 and the total number of elements in the edge vector, which is 6, are generated. Let us assume that the numbers generated are 3 and 6. Since the 3rd and 6th elements of the edge vector are node indices 2 and 3, the new node 4 will connect to the existing nodes 2 and 3 as shown in Figure 6. The updated connection matrix and edge vector due to the new node connection are shown in equations (8), (9), (10), and (11). From the updated edge vector, we can see that the number of node indices 2 and 3 has increased from 2 to 3. Thus, nodes 2 and 3 have higher probability of being selected by the new node 5 compared to nodes 1 and 4. The dynamic addition of a new node and preferential attachment processes are implemented based on three blocks: new node \(t\) generation block, \(m\) target node selection block, and connection matrix/edge vector update block, as shown in Figure 5. When a network construction with \(N = t + m_0\) nodes and \(mt\) edges has been completed, the final node connection matrix is used in the metric calculation blocks: average path length block and degree distribution block.

5.3. Metric Calculation Modules. In average path length calculation block, after completion of a scale-free network construction, the node connection matrix that contains the link information between all the nodes is used for average path length calculation. The steps used in the average path length calculation block for small-world network are also used in the scale-free network system model. Furthermore, the BFS algorithm is recommended for average path length calculation of scale-free networks with a large number of nodes and edges.

Degree distribution calculation is an important factor in studying the power-law nature of the constructed scale-free network. In degree distribution calculation block, the first step is to gather degree \(K\) for all the nodes in the network using node connection matrix. The range of the degree is defined based on the minimum and maximum degree values found in the first step. The next step is to bin the degree data covering all the degree range. In the last step, the power-law nature of the network is verified by plotting the binned degree data in log-log scale.

6. IoT Model

In contrast to the relational graphs used in general complex network models, spatial graphs are more appropriate models for wireless sensor networks (WSNs). This is because in WSN, due to practical constraints, such as energy capacity and radio transmission range, the links are restricted by the distance between nodes, rather than relational factors as in small-world networks and scale-free networks. Thus, a new incoming node will have a limited number of candidate target nodes to be connected subject to required power consumption to communicate between nodes \(i\) and \(j\) that can be simply defined as \(p_{ij} = p_{\text{min}}d_{ij}^{-\gamma}\), where \(\gamma\) is the path loss exponent and \(p_{\text{min}}\) is the minimum power for acceptable reception quality. Furthermore, in heterogeneous WSN, the nodes will have different communication and energy capabilities. Possible WSN nodes are a sink node, a small number of cluster head nodes with high hardware capabilities and energy capacity, and a large number of low cost sensor nodes that are continuously added to the network over time.

Important optimization criteria of WSN are energy efficiency, average path length based on geographical distance, and network tolerance. Based on these WSN optimization criteria, the rewiring scheme in the conventional small-world network algorithm is modified to include performance metric such as energy efficiency as shown below.

Step 1. Start with a ring of \(N\) nodes.

Step 2. Connect \(K\) nearest nodes for all the nodes \(i = 1 \cdots N\).

Step 3. Reconnect or add new edges based on probability function that is dependent on performance optimization parameters.

Step 4. Repeat Step 3 for all \(NK/2\) edges in the ring network.

As for the WSN based on scale-free properties, the preferential attachment in the conventional scale-free network
algorithm is modified to include performance metric such as energy efficiency as shown below.

**Step 1.** Start with a small number \( m_0 \) nodes with degree \( K \).

**Step 2.** Introduce a new node into the network.

**Step 3.** Connect the new node to \( m \) existing nodes based on probability that will maximize performance optimization criteria.

**Step 4.** Repeat Steps 2 and 3, until a network with \( N = t + m_0 \) nodes and \( mt \) edges has been constructed.

### 7. Simulation Results

#### 7.1. Small-World Network.

In this section, we study the behavior of the small-world network implemented based on the system architecture with metric calculation modules described in Section 4. We initially assumed a regular ring lattice model with \( N = 24 \) nodes and initial degree \( K = 4 \) for all the nodes. The small-world network was created according to the system architecture described previously with various rewiring probability \( p \) ranging from 0 to 1. Figure 7 shows the average path length of the implemented small-world network. One could observe that the average path length is around 4.2 for \( p = 0 \) (without rewiring) and decreases to 2.6 for high rewiring probability \( p \) (random network). Even with small number of random rewiring, there is a drastic decrease in average path length. Note that the theoretical average path length value for random network \( (p = 1) \) can be calculated as \( L \sim \ln(N)/\ln(K) \). Figure 8 shows the clustering coefficient of the implemented small-world network. One could observe that the clustering coefficient remains relatively constant with value around 0.5. However, there is rapid drop in the clustering coefficient for rewiring probability \( p \) greater than 0.1. Thus, we can observe that the small-world network remains highly clustered like regular lattice for \( p \) less than 0.1. From Figures 7 and 8, we can conclude that the behavior of the small-world network was fully confirmed, having highly clustered behavior as the regular lattice and small average path length as the random graphs, based on the proposed system architecture.

#### 7.2. Scale-Free Network.

In this section, we study the behavior of the scale-free network implemented based on the system architecture described in Section 5. Figure 9 compares the average path length for random networks, denoted as RN in the plot, and scale-free networks, denoted as SFN in the plot, for different number of nodes \( N \) in the network with \( \langle K \rangle = 4 \). The random network was constructed based on the algorithm described in Section 3. As for the scale-free network, the initial number of nodes was set to \( m_0 = 2 \) and the number of target nodes for preferential attachment by the incoming node was set to \( m = m_0 = 2 \). To calculate the average path length, the BFS algorithm was utilized. As shown in Figure 9, with increase in network size \( N \), substantial decrease in average path length in scale-free network is observed compared to a random network. Figure 10 shows
the degree distribution of a scale-free network with total number of nodes in the network equal to $N = 500$ for different number of initial nodes and target nodes set as $m_0 = m = 3, 5, 7$. The degree distribution was calculated using the degree distribution calculation block described in Section 5. As seen in Figure 10, the degree distribution generated by the proposed system model follows the power-law distribution for all different values of $m_0$ and $m$ proving that the generated network evolves into a scale-free network. Note that the noise in the tail occurs due to limited number of data to average out the noise. One of the advantages of the scale-free network is the error and attack tolerance. Figure 11 shows the error tolerance performance as a function number of nodes removed from the network. To study the error tolerance performance of the generated scale-free network, out of $N = 1000$ nodes, randomly selected nodes were removed with removal of all the connected edges to that node. Average path length metric was used to study the disruption effect to the scale-free network and random network due to removal of nodes. We can see that when 20% of the nodes in the network are removed, the average path length of the network increases around 16% and 12% in random network and scale-free network, respectively. Note that a peak point can be observed from the figure. This point is called a critical point where the network breaks into numerous isolated clusters, resulting in rapid drop in average path length. From the small average path length, power-law degree distribution, and error tolerance performance shown in Figures 9, 10, and 11, we can conclude that the generated network fully satisfies the scale-free characteristics.

8. Conclusions
In this paper, we introduced basic concepts of complex networks, including small-world networks and scale-free networks. Then, separate system models for small-world networks and scale-free networks were proposed that can be easily implemented and applied to IoT network optimizations. The small-world networks model contains five major blocks: node initialization block, node connection block, rewiring block, average path length calculation block, and clustering coefficient calculation block. As for the scale-free networks model, major blocks are node initialization block, new node generation block, target node selection block, average path length calculation block, and degree distribution calculation block. The proposed system models were evaluated based on various complex network metrics. The simulation results show that one can confirm the small-world characteristics showing highly clustered behavior and small average path length based on the proposed system architecture. Furthermore, the degree distribution generated by the proposed system model followed the power-law distribution for all different values of $m_0$ and $m$ proving that the generated network evolves into a scale-free network.

Competing Interests
The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgments
This work was supported by the Dongguk University Research Fund.

References
[1] A. Meddeb, “Internet of things standards: who stands out from the crowd?” IEEE Communications Magazine, vol. 54, no. 7, pp. 40–47, 2016.
[2] V. P. Kafle, Y. Fukushima, and H. Harai, “Internet of things standardization in ITU and prospective networking technologies,” IEEE Communications Magazine, vol. 54, no. 9, pp. 43–49, 2016.
[3] M. E. Newman, Networks: An Introduction, Oxford University Press, Oxford, UK, 2010.
[4] S. Milgram, “The small world problem,” Psychology Today, vol. 1, no. 61, 1967.
[5] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’ networks,” Nature, vol. 393, no. 6684, pp. 440–442, 1998.
[6] X. F. Wang and G. Chen, “Complex networks: small-world, scale-free and beyond,” IEEE Circuits and Systems Magazine, vol. 3, no. 1, pp. 6–20, 2003.
[7] L. A. N. Amaral, A. Scala, M. Barthelemy, and H. E. Stanley, “Classes of small-world networks,” Proceedings of the National Academy of Sciences of the United States of America, vol. 97, no. 21, pp. 11149–11152, 2000.
[8] J. M. Kleinberg, “Navigation in a small world,” Nature, vol. 406, no. 6798, p. 845, 2000.
[9] J. M. Kleinberg, “The small-world phenomenon: an algorithmic perspective,” in Proceedings of the ACM Symposium on Theory of Computing, pp. 163–170, 2000.
[10] M. E. J. Newman, “Models of the small world,” Journal of Statistical Physics, vol. 101, no. 3–4, pp. 819–841, 2000.
[11] A. Barrat and M. Weigt, “On the properties of small-world network models,” The European Physical Journal B—Condensed Matter and Complex Systems, vol. 13, no. 3, pp. 547–560, 2000.
[12] I. Sohn, “Modeling and implementing small world network,” in Proceedings of the International Conference on Information and Communication Technology Convergence (ICTC ’06), pp. 414–416, IEEE, Jeju Island, South Korea, October 2016.
[13] A.-L. Barabási and R. Albert, “Emergence of scaling in random networks,” Science, vol. 286, no. 5439, pp. 509–512, 1999.
[14] A.-L. Barabási, R. Albert, and H. Jeong, “Mean-field theory for scale-free random networks,” Physica A: Statistical Mechanics and its Applications, vol. 272, no. 1–2, pp. 173–187, 1999.
[15] R. Albert and A.-L. Barabási, “Statistical mechanics of complex networks,” Reviews of Modern Physics, vol. 74, no. 1, pp. 47–97, 2002.
[16] A.-L. Barabási, E. Ravasz, and T. Vicsek, “Deterministic scale-free networks,” Physica A, vol. 299, no. 3–4, pp. 559–564, 2001.
[17] R. Cohen and S. Havlin, “Scale-free networks are ultrasmall,” Physical Review Letters, vol. 90, no. 5, Article ID 058701, 2003.
[18] R. Albert and A.-L. Barabási, “Statistical mechanics of complex networks,” Reviews of Modern Physics, vol. 74, no. 1, pp. 47–97, 2002.
[19] B. Bollobás and O. M. Riordan, “Mathematical results on scale-free random graphs,” in Handbook of Graphs and Networks: From the Genome to the Internet, pp. 1–34, 2003.
[20] L. Zhao, K. Park, and Y.-C. Lai, “Attack vulnerability of scale-free networks due to cascading breakdown,” Physical Review E—Statistical, Nonlinear, and Soft Matter Physics, vol. 70, Article ID 035101, 2004.
[21] A.-L. Barabási, “Scale-free networks: a decade and beyond,” Science, vol. 325, no. 5939, pp. 412–413, 2009.
[22] A. Helmy, “Small worlds in wireless networks,” IEEE Communications Letters, vol. 7, no. 10, pp. 490–492, 2003.
[23] D. L. Guidoni, A. Boukerche, H. A. B. F. Oliveira, R. A. F. Mini, and A. A. F. Loureiro, “A small world model to improve synchronization algorithms for wireless sensor networks,” in Proceedings of the 15th IEEE Symposium on Computers and Communications (ISCC ’10), pp. 229–234, IEEE, Riccione, Italy, June 2010.
[24] T. Zhang, J. Cao, Y. Chen, L. Cuthbert, and M. Elkashlan, “A small world network model for energy efficient wireless networks,” IEEE Communications Letters, vol. 17, no. 10, pp. 1928–1931, 2013.
[25] Y.-B. Kim, B. Hong, and W. Choi, “Scale-free wireless networks with limited degree information,” IEEE Wireless Communications Letters, vol. 1, no. 5, pp. 428–431, 2012.
[26] Y. Wang, E. Liu, Y. Jian et al., “NLH: a complex network model with compensation for enhanced connectivity,” IEEE Communications Letters, vol. 17, no. 9, pp. 1856–1859, 2013.
[27] D. Wang, E. Liu, Z. Zhang et al., “A flow-weighted scale-free topology for wireless sensor networks,” IEEE Communications Letters, vol. 19, no. 2, pp. 235–238, 2015.
[28] D. Luo, T. Qiu, N. Deonauth, and A. Zhao, “A small world model for improving robustness of heterogeneous networks,” in Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP ’15), pp. 849–852, IEEE, Orlando, Fl, USA, December 2015.
[29] Z. Dong, Z. Wang, W. Xie, O. Emelumadu, C. Lin, and R. Rojas-Cessa, “An experimental study of small world network model for wireless networks,” in Proceedings of the 36th IEEE Sarnoff Symposium (Sarnoff ’15), pp. 70–75, IEEE, Newark, NJ, USA, September 2015.
[30] Y. Jian, E. Liu, Y. Wang, Z. Zhang, and C. Lin, “Scale-free model for wireless sensor networks,” in Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC ’13), pp. 2329–2332, IEEE, Shanghai, China, April 2013.
[31] H. Peng, S. Si, M. K. Awad, N. Zhang, H. Zhao, and X. S. Shen, “Toward energy-efficient and robust large-scale WSNs: a scale-free network approach,” IEEE Journal on Selected Areas in Communications, vol. 34, no. 12, pp. 4035–4047, 2016.
[32] D. G. Corneil, “Lexicographic breadth first search—a survey,” in Graph-theoretic concepts in computer science, vol. 3353 of Lecture Notes in Comput. Sci., pp. 1–19, Springer, Berlin, Germany, 2004.