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

¹*Electronics and Telecommunications Research Institute, Daejeon, Republic of Korea*

²*Ajou 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

FIGURE 1: Architectural illustration of the DWSAN system.

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 multitiered, 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 blinders 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,

FIGURE 2: Two versions of our custom sensing platform in the DWSAN system. The platform includes temperature/humidity sensors, illumination (light) sensors, CO and CO₂ sensors, and a PIR sensor.

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 CO₂ 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, multitiered 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

FIGURE 3: Pictures of DWSAN's actuation platforms. (a) shows picture of an application driven light control actuation unit, and (b) shows our generic electricity controlling actuation unit for 220 V devices and (c) shows the inner details.

FIGURE 4: Picture of our master gateway node in our DWSAN system.

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-correlation algorithm. 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

FIGURE 5: Illustration of the server architecture in the DWSAN system.

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.

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

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 smoothen some of the dynamics, we apply this filter so that small variations in the data will not trigger reports. Next, we compare the short term mean of the recent measurements, sample , with the most recent measurement report sent to the actuator, $\text{sample}_{\text{act}}$. The difference, Δ , is computed and compared once again with a predefined threshold, Δ_{TH} . Δ_{TH} can be configured by the profile messages with respect to the data analysis algorithm's tolerable error-bounds for the summary reports. If $\Delta > \Delta_{\text{TH}}$, we update $\text{sample}_{\text{act}}$ as sample and report sample to the actuator. If $\Delta \leq \Delta_{\text{TH}}$, since the current short term mean of the samples is within the predefined error-bounds, no reporting takes place. Using this simple scheme compliments the DWSAN architecture in minimizing the number of packets to transmit.

In addition, when no data points exceed the threshold, DWSAN sensor nodes also trigger reports using an multiplicatively increasing/decreasing trickle timer [17]. The timer's interval is increased multiplicatively on a periodic-basis until reaching an upper limit. At this upper limit the timer will stay at its upper state until an event occurs to reset the timer to a smaller value. At the beginning of each interval, nodes select a random time within $[\text{current time}, \text{current time} + \text{interval}]$ to issue a measurement report. Even if reports are sent before the interval is over, the timer waits for the end of the interval to update the interval and then selects the next report time. When the Δ -based event detection scheme indicates that a report is sent (e.g., detected changes in the data), the timer's interval is decreased multiplicatively to send more reports to accurately track the dynamic periods while suppressing reports when there are only minimal changes in the data. Note that here, unlike the original trickle timer

FIGURE 7: Sensing-to-actuator decision and response latency for DWSAN and a centrally managed network.

design, instead of resetting the timer value upon events to the minimal, we take a more conservative approach of multiplicatively decreasing the timer value.

While this adaptive sampling functionality is optional (there is no operational issue to the DWSAN architecture when not using it) applications such as temperature and humidity management, or any application that deals with data with similar characteristics, can benefit significantly from using such sensor data suppression techniques.

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

FIGURE 8: Number of packets that are transmitted for the centralized approach and DWSAN.

medium) was set to ~ 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, eventually 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 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 $\times$ 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 Δ_{TH} . Quantitatively speaking, the original sample set consisted of ~ 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 $\Delta_{TH} = 1.0^\circ\text{C}$. 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

FIGURE 9: Percentage of sample points required to reconstruct a per-second sample stream when using our proposed adaptive sample scheme with varying target thresholds.

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 1pps, 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 more 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 $\sim 50\%$ compared to a IETF-RPL-based single-tier network architecture [12]. Deeper investigation on the traces revealed that $\sim 10\%$ 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 $\sim 65\%$ 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

FIGURE 10: Mean error of our proposed adaptive sampling scheme to the original data compared to periodic schemes with same sample counts.

FIGURE 11: Data summary from our proposed scheme compared to a periodic scheme with the same number of samples.

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.

FIGURE 13: Internal deployment at research-scale.

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, "Code-Blue: 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.
- [18] K. JeongGil, P. Jongjun, A. Jong Jun, and K. Naesoo, "Just send me the summary: analyzing sensor data for accurate summary reports in indoor environments," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems (SenSys '12)*, pp. 325–326, ACM, New York, NY, USA, 2012.
- [19] H. Park, J. Park, H. Kim et al., "ReLiSCE: utilizing resource-limited sensors for office activity context extraction," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 8, pp. 1151–1164, 2014.
- [20] E. Cañete, J. Chen, M. Díaz, L. Llopis, and B. Rubio, "A service-oriented approach to facilitate WSN application development," *Ad Hoc Networks*, vol. 9, no. 3, pp. 430–452, 2011.
- [21] Y. Wu and A. Rowe, "Logic-based programming for wireless sensor-activator networks," in *Proceedings of 2011 IEEE/ACM 2nd International Conference on Cyber-Physical Systems (ICCPs '11)*, pp. 163–173, Chicago, Ill, USA, April 2011.

Hindawi

Submit your manuscripts at
<https://www.hindawi.com>