Research Article

Priority-Based Dynamic Multichannel Transmission Scheme for Industrial Wireless Networks

Yuichi Igarashi,¹ Yoshiki Matsuura,¹ Minoru Koizumi,¹ and Naoki Wakamiya²

¹*Hitachi, Ltd., Research & Development Group, Center for Technology Innovation, Tokyo, Japan*

²*Osaka University, Osaka, Japan*

Correspondence should be addressed to Yuichi Igarashi; yuichi.igarashi.hb@hitachi.com

Received 31 March 2017; Accepted 15 August 2017; Published 27 September 2017

Academic Editor: Giovanni Pau

Copyright © 2017 Yuichi Igarashi 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.

Industrial wireless sensor network (IWSN) applications are required to provide precise measurement functions as feedback for controlling devices. Current industrial wireless communication protocols, such as ISA100.11a and wirelessHART, have difficulty, however, in guaranteeing latency for unpredictable on-demand communications. In this paper, a priority-based dynamic multichannel transmission scheme is proposed for IWSNs. In the proposed scheme, a root node controls the transmission timing of high-priority packets, while other nodes autonomously decide what channel to use and when to transmit packets to a neighbor. Simulation results show that real time control is possible where a response delay from transmission of a request to reception of a reply at a root node is within 1,140 ms at per-link communication success probability with a retry of higher than 93%.

1. Introduction

Industrial wireless sensor networks (IWSNs) have been emerging as a new means of communication for social infrastructure applications like Advanced Metering Infrastructure (AMI), Distribution Automation, Optimized Factory, Predictive Maintenance, Building Automation, and so on. These applications basically gather information from remote devices, that is, sensors, check device status or circumstance, and control devices, that is, actuators, based on the gathered information. For remote monitoring devices, the main purpose is periodic collection of device status or sensor data. At the same time, industrial applications also require on-demand communication for data collection and operation of devices by a remote control server, within specific end-to-end deadlines. For instance, an AMI system requires a deadline of 20–60 seconds when a remote control server requests on-demand meter reading. Although such request/reply type of communication is unpredictable, IWSNs must guarantee a maximum communication delay for both periodic and unpredictable packets.

Wireless networks using legacy Media Access Control (MAC) protocols based on Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA), such as WiFi or Zigbee [1],

sometimes drop packets because of interference from other wireless networks. To increase the reliability of wireless networks for industrial applications such as remote monitoring devices, several protocols, such as Wireless HART [2], ISA100.11a standard [3], and IEEE802.15.4e [4], have been developed and standardized.

These standards use Time Division Multiple Access (TDMA) and multiple channel hopping technologies for their MAC protocols. Both WirelessHART and ISA100.11a for industrial applications support a centralized network management architecture in which all nodes in an IWSN are time-synchronized and assigned communication timings for periodic data gathering by a central network manager. It decreases both interference within the IWSN and interference from other wireless networks using the same channels, thus increasing the rate of successful communication.

However, high success rate of packet transmission of periodic communication alone is not sufficient for industrial applications. Assume that an on-demand downward packet from a root node to a sensor and upward packets from sensors to the root node are generated on the same path in a multihop TDMA-based wireless network. They compete for a channel and an opportunity of packet transmission and thus some packets will have to wait until the next transmission

opportunity. This may cause quite large accumulated end-to-end delay, making it difficult to guarantee a maximum delay to applications. To mitigate delay, one of solutions is to assign transmission timings to all nodes for unpredictable communication. However, because of unpredictability, such assignment is very likely to be redundant and as a result precious network bandwidth will be considerably wasted.

To guarantee a deadline for such unpredictable packets, we propose a priority-based dynamic multichannel transmission scheme. Our scheme prioritizes packets in accordance with application requirements. Packet transmission is scheduled in a slotted manner, but detailed slot allocation as in usual TDMA-based protocols is not performed. More specifically, a root node only determines both when it transmits on-demand request packets for remote control and when nodes transmit data packets for periodic monitoring. On the other hand, packet forwarding is not scheduled at all. In a slot, which we call SlotFrame, priority CSMA/CA-like packet transmission is performed at each node. Our scheme operates over a MAC layer and does not rely on any specific MAC protocol. We discuss compatibility with ISA100.11a in Section 6.2.

We consider one type of packets for periodic communication and three types of packets for unpredictable communication. We first define three priorities of packets depending on their type and then assign one dedicated channel to each priority. The highest priority is given to unpredictable and on-demand packets that a root node sends to a node for control or information retrieval, that is, downward packets. Upward reply packets are also given the highest priority, because request/response communication for device control requires a very short end-to-end delay. The second priority is given to periodic packets used for regular data collection. Network control packets are then set to the lowest priority. In our proposal, a root node can thus transmit the highest-priority packets at any time but still control the transmission timing through centralized administrative control, as in ISA100.11a. More specifically, a node replies to a request packet at the time specified by the root node.

On the other hand, each node decides the time of forwarding a packet by autonomous decentralized radio channel control. A node scans three communication channels in descending order of priority and dynamically decides which channel to use. For example, when a node having a reply packet finds that a request packet is to be sent by a neighbor, it defers transmission of the packet for a certain duration of time to avoid collision among downward and upward packets over the high-priority channel. If there is no transmission of high-priority packets in the vicinity, a node can transmit a periodic data packet using another channel for the middle priority. Only when there is no packet transmission on either high or middle priority channel, a node can transmit a network control packet. This mechanism ensures that the highest-priority packets are preferentially transferred without unexpected delay.

The contribution of this paper is proposing a priority-based multichannel transmission scheme that determines when and what packets should be transmitted on which channel. Through simulation, we validate our proposal for

TABLE 1: Application and system requirements.

	920-MHz band	2.4-GHz band
Applications	(i) AMI (ii) Distribution automation	(i) Optimized factory (ii) Predictive maintenance
Application processing	(i) Remote monitoring (ii) Remote operation	(i) Remote monitoring (ii) Remote operation
Communication type	(i) Publish/subscribe (ii) Request/response	(i) Publish/subscribe (ii) Request/response
Remote monitoring cycle	30 min (periodic data)	1–5 min (periodic data)
Maximum delay for remote operation	20–60 sec (unpredictable data)	5–20 sec (unpredictable data)
Number of nodes in wireless network	1–500 nodes	1–500 nodes
Packet length	500–600 B	90 B
Communication speed	50–100 kbps	250 kbps
Protocols	(i) IEEE802.15.4g/e (ii) 6tisch (iii) 6LoWPAN	(i) ISA100.11a (ii) WirelessHART

two industrial applications: AMI and industrial process monitoring and control. We also theoretically estimate the lower bounds of available bandwidth for middle- and low-priority packet transmission. Moreover, TDMA-based protocols like ISA100.11a typically have a scheduler for allocating network resources such as time slots to all nodes. This scheduling process is often complicated and the scheduler has to deliver the information to all nodes whenever a new node joins the network or a network topology changes. In contrast, since our scheme only determines when to generate and transmit a packet at a root node for remote control and at nodes for periodic monitoring, it does not need to adjust a schedule as far as the maximum number of hops and the number of nodes do not change. We discuss this advantage in more detail in Section 6.3.

The rest of the paper is structured as follows. We first describe requirements and challenges in Section 2, and Section 3 presents an overview of related work. In Section 4, we propose the priority-based transmission scheme. Then, we evaluate the communication delay for highest-priority packets and available bandwidth for other packets in Section 5. In Section 6, we discuss compatibility with ISA100.11a and overhead incurred in implementing our proposal before discussing our conclusions and future work in Section 7.

2. Requirements and Challenges

2.1. System Requirements from Industrial Applications. Table 1 lists a summary of industrial applications and system requirements for IWSN [5–9]. AMI, mentioned in Section 1, has been intensively deployed in Japan since the Great East Japan

FIGURE 1: Target system configuration.

Earthquake in 2011. Industrial manufacturing companies have recently faced a need to increase productivity and optimize factories. To achieve this, IWSN systems require a variety of data and means of field data collection, like sensor data from factories and buildings, smart meters, and so on. To collect such large data, standard Internet of things (IoT) wireless technologies have provided large, dense wireless networks that contain several hundred nodes and form at most 8-16 multihop networks. However, collection of periodic data with high success in communication is necessary [7, 8].

Moreover, unpredictable and on-demand communication occurs in an IWSN. For example, eco-friendly systems such as distribution automation and demand response systems have been deployed in order to use energy efficiently. These systems request communication of contingency packets from a remote server to an end device in an unpredictable way. Industrial applications such as industrial process control [5] also require controlling devices remotely with a maximum delay of several seconds. Figure 1 shows an example of a target system configuration for the application of monitoring and controlling a building.

2.2. Challenge. The packet error rate (PER) is normally one of the most important parameters for evaluating reliability of a wireless network. In addition, guaranteed deadlines should be considered for IWSNs, because industrial applications require real time processing. Several wireless protocols have been already standardized and developed for industrial applications, including WirelessHART and ISA100.11a. To decrease the PER for collecting periodic data from sensors in dense and lossy wireless networks, these standards use TDMA-based MAC protocols. Such protocols overcome the problem of packet collisions in the network. In fact, the data collection ratio for WirelessHART reaches more than 99%, because SmartMesh WirelessHART devices basically perform retransmission twice at most [10]. This performance seems sufficiently high for remote monitoring purposes.

On the other hand, when an unpredictable on-demand packet has to be sent to or from a node, the packet either consumes assigned bandwidth or waits several seconds until the next assigned bandwidth becomes available. This can

cause random latency, which further depends on TDMA scheduling, retransmission timing, and wireless radio conditions. Most WSNs do not support real time communication [11] and it is difficult for even TDMA-based MAC protocols such as IEEE802.15.4e to support real time communication for large scale networks like AMI [12].

As described below in Section 3, when we use normal ISA100.11a for both remote monitoring and remote control of devices, it does not guarantee the latency of on-demand and multihop communication at any time, although it can transfer a higher-priority packet by applying the priority CSMA/CA scheme among single-hop neighbors. If such on-demand communication can be expected, then a system manager with an optimized scheduler may enable ISA100.11a to allocate communication timing for all nodes in order to transfer a higher-priority packet within a certain delay and with due consideration to maintain a high data collection ratio. Whenever the network topology is changed, because of the instability of the radio environment, however, the schedule must be updated, so this is an unrealistic solution.

As another solution to the problem, we could use multiple ISA100.11a network functions on different network interfaces, that is, one ISA100.11a function for remote monitoring and another for remote control, for example. In this case, each function would concentrate on scheduling transmission of a single application packet with a high end-to-end path success probability. Unfortunately, this approach faces the same problem that a system manager must deliver an optimized schedule to all nodes whenever the network topology is changed. Moreover, each node would have to control multiple ISA100.11a network functions precisely, but the standard does not describe how an application can manage multiple ISA100.11a networks.

Therefore, our challenges in this paper are to mitigate unexpected latency for unpredictable and high-priority IWSN communications and to show how to meet system requirements for high end-to-end success probability of periodic communication.

3. Related Work

In wireless sensor networks, MAC is a key technology that determines channel access delay and utilization. MAC protocols are roughly classified into three types: contention-based, contention-free, and hybrid.

First, contention-based schemes (using CSMA/CA) such as IEEE802.15.4 determine transmission timing by checking existence of carrier signals, that is, carrier sense. When a network is large or dense, the PER is normally high and as such CSMA/CA-based MAC protocols cannot guarantee latency [13-15].

Second, contention-free MAC protocols using TDMA implement scheduled communication with a centralized coordinator, such as a network manager. In TDMA-based MAC protocols, a node transmits and forwards a packet to a neighbor according to an allocated time slot schedule. When packet transmission fails, a node should wait until the next assigned time slot to resend the packet. Therefore, the end-to-end delay depends on the whole schedule and its cycle

FIGURE 2: A shared time slot using the CSMA/CA technology of ISA100.11a.

length called superframe. To reduce latency in industrial networks, Saputra and Shin proposed a scheduling scheme for ISA100.11a superframes [16]. This scheme specifies how to build a superframe to guarantee the delay for periodic upward packets from sensors to a root node and how to check the schedulability of a superframe.

Finally, IWSNs often adopt hybrid schemes [2, 3, 17, 18]. While the hybrid standard schemes such as ISA100.11a and WirelessHART use a TDMA-based MAC protocol, they also provide periodic data communication at low PER. At the same time, these schemes adopt a CSMA/CA-based MAC protocol for unpredictable transmission requirements, such as network control packets, alert information, on-demand requests, and retransmission of data packets.

During a CSMA/CA period in a hybrid scheme, ISA100.11a nodes can use a priority CSMA/CA scheme, as shown in Figure 2. Waiting time proceeding to transmission of a high-priority packet is shorter than that of a low-priority packet as shown in Figure 2, where transmitter 1 has a high-priority packet and transmitter 2 has a low-priority one. Because of difference in waiting time, transmitter 2 can detect transmission of a high-priority packet during its CCA (Clear Channel Assessment) and stop the transmission attempt. This scheme enables priority control within single-hop communication and decreases the probability of collision among transmission of packets of different priority [19]. We also use this priority CSMA/CA-like scheme in our approach.

4. Priority-Based Transmission Scheme with Dynamic Channel Shift

In this section, we provide assumptions and terminologies at first. Then, we present an outline of our priority-based transmission scheme with dynamic channel shift. We also give detailed algorithms for priority-based channel selection and the transmission and reception mechanisms.

4.1. Assumptions

Industrial Applications Features. As noted above, our target applications are AMI, Distribution Automation, Optimized Factory, and so on. These typical industrial applications

normally collect field data and store them at a remote server like cloud. Industrial applications often involve real time communication to react or respond to user queries within a predetermined deadline in order to perform timely control and avoid failures. Since most of existing WSNs cannot satisfy the requirement, we propose a scheme to provide real time communication over a large scale IWSN.

MAC Protocols. Recently, many new industrial wireless systems have been deployed. Most of them use TDMA-based protocols such as ISA100.11a, wirelessHART, and IEEE802.15.4e/g based protocols in order to avoid interference among internal nodes and keep communication success probability high. Those protocols provide multihop and time-synchronized networks that consist of a central manager and other nodes that are synchronized with the central manager. We assume that our targeted network is also multihop and time-synchronized, but our proposal operates over a MAC layer to decide what channel to use and when to transmit packets and does not rely on any specific MAC protocols.

Network Topology and Its Condition. Similarly to other TDMA-based WSN protocols, we also assume a tree topology whose root is a central manager. Our proposal does not specify any routing protocols as far as a stable tree-based routing topology is established and maintained for a large scale WSN. In simulation experiments, we consider a network of 500 nodes with 8 hops for 920 MHz and 16 hops for 2.4 GHz at maximum.

Priority Level of Packets. In an IWSN, multiple applications would simultaneously operate such as periodic data gathering and remote control. In addition, networking functions such as routing and time synchronization are also running. Among them, remote control is the most crucial and must be given the highest priority to guarantee real time communication. Furthermore, its responses from nodes to a root node should have the higher priority than those packets belonging to periodic data gathering. Although frequent loss of control packets affects stability and reliability of a WSN, a best-effort service is enough. We evaluate the lower bound of available bandwidth for lower priority packets in Section 6. Details of prioritization will be given in the next subsection.

FIGURE 3: A composition of a SlotFrame.

FIGURE 4: A composition of a ComFrame.

Multiple Communication Channels. TDMA-based MAC protocols for IWSNs have channel hopping functions to enable coexistence of multiple networks in the same area and dynamic bandwidth allocation. In this paper, we assume that three channels are available to use.

4.2. Terminologies. We define our terminologies as follows.

4.2.1. Frame Composition. First, we define three types of frames over a MAC protocol.

The first type is the *SlotFrame*, which consists of two timeslots as shown in Figure 3. Both ISA100.11a and IEEE802.15.4e technologies divide time into timeslots of configurable length, with typical durations ranging from 10 to 14 ms. These technologies do not, however, support MAC layer retransmission within a timeslot. The *SlotFrame* enables transmission of a packet within the 1st timeslot and retransmission within the 2nd timeslot. ΔL_{SF} denotes the length of a *SlotFrame*, for example, 20 to 28 ms.

The second type is the *ComFrame*, which consists of *SlotFrames*, as shown in Figure 4. The number of *SlotFrames* in a *ComFrame* is calculated as the maximum hop count in a multihop wireless network plus 3. In ISA100.11a, a root node knows the whole network topology. The number “3” is a key number that was chosen to avoid hidden terminal problems on a multihop route, as described in detail later. In this paper, we assume that the number of *SlotFrames* in a *ComFrame* is 11 (8 hops + 3). In addition, through centralized administrative control, a root node assigns a *ComFrame* to a node when it joins the network. The assignment does not change even if the network topology changes.

The final frame type is the *AppFrame*, which consists of *ComFrames* on multiple channels, as shown in Figure 5. The number of *ComFrames* in an *AppFrame* is a system parameter that depends on the application requirements. ΔL_{AF} denotes the length of a *AppFrame*. For example, if an application remotely operates all devices in 30 min and collects data from all devices in 30 min, then $\Delta L_{AF} = 30$ min. In Figure 5, the network manager divides the *AppFrame* to two blocks. In this example, block 1 is used for controlling

all nodes (*L1* and *L2* packets), collecting data from all nodes (*L3* packets), and transmitting network control packets (*L4* packets). Other blocks, such as block 2 in Figure 5, are used for bidirectional communication (*L1* and *L2* packets) needed for repeat attempts at remote control or data collection from devices, and for transmitting network control packets (*L4* packets). A system manager determines the number of blocks in an *AppFrame*.

4.2.2. Priority Level and Communication Channels. As mentioned above, we define 4 priority levels. The highest level (*L1*) is for downward packets from a root node to a sensor node (end device) that an application controls. The second (*L2*) is for upward packets in response to *L1* packets. The third (*L3*) is for periodically collected data transferred from an end device to a root node, for example, a network health report or sensing data. The lowest priority level (*L4*) is for network control packets, for example, routing packets, time synchronization packets, or beacon packets. In our proposed scheme, *L1* and *L2* packets are transferred over communication channel 1 (Ch1), *L3* packets are transferred over channel 2 (Ch2), and *L4* packets are transferred over channel 3 (Ch3). Here, Ch1 and Ch2 are contention-free channels like TDMA-based communication, whereas Ch3 is a contention-based channel like CSMA/CA-based communication.

4.3. Outline. We give an outline of how our scheme simultaneously fulfills several requirements of industrial wireless communications: a guaranteed deadline for on-demand communication, data collection at low PER, and communication of network control packets among neighbors.

In our scenario, there are three kinds of packets. The first kind is unpredictable packets for on-demand control. The second is periodic packets generated by sensors for periodic data collection. The third is network control packets that build multihop routes from sensors to a root node and exchange time information for synchronization among nodes.

We first rank packets according to industrial application requirements. To provide a guaranteed deadline, we define an on-demand downward packet from a root node to a sensor to have the highest priority (*L1*) and an on-demand upward reply packet from a sensor to a root node to have the second-highest priority (*L2*). The third priority (*L3*) is for periodic data collection packets from any sensor, and the lowest priority (*L4*) is for network control packets.

In addition, our priority-based dynamic multichannel transmission scheme uses three communication channels. The *L1* and *L2* packets between a root node and sensor nodes share a communication channel (Ch1). The periodic *L3* packets use another communication channel (Ch2) for a certain period of time, and the *L4* packets use a third channel (Ch3). In other words, a root node sends an on-demand request packet with priority *L1* while waiting to receive a reply packet (*L2*) for a previous request packet. During the same period, a sensor node sends a periodic data collection packet (*L3*) to a root node on Ch2. The timing for a sensor node to transfer such a periodic packet to a root node is decided by the

FIGURE 5: A composition of an AppFrame.

FIGURE 6: A simple example of network topology.

root node when the sensor node joins the network. Network control packets (L4) should be transferred only when no neighbors have to transfer higher-priority packets. In each SF, nodes scan channels in order of priority. When a node detects any packet is being transmitted in a higher-priority channel, it stays at the channel and receives the packet. Otherwise, it moves to a lower priority channel and checks the existence of packets. We describe details later in this section.

4.4. Example of Priority-Based Dynamic Multichannel Transmission Mechanism. We next provide an example of how to ensure preferential communication of a downward packet from a root node to an end device (L1) and how to avoid contention between a downward packet and an upward packet (L2) in response to a previous downward packet. As noted above in Sections 4.1 and 4.3, the order of the priority is predetermined and all nodes share the information. Figure 6 shows a simple example of a network topology. The network consists of 8 nodes and has a maximum of 4 hops. Figure 7 shows an example of packet flow, in which the root generates an L1 packet and node 3 generates an L2 packet. At the 2nd

FIGURE 7: An example of packet flow over highest priority channel (Ch1).

SlotFrame in the ComFrame, node 2 cancels forwarding of the L2 packet to node 1. Then, node 2 waits two SlotFrames to avoid collisions due to the hidden terminal problem. While node 2 waits, the L1 packet is delivered to node 4 without any delays. Then, the L2 packet is eventually transferred to the root node at the 6th SlotFrame. This crossed-transfer mechanism guarantees a maximum latency for transmission of the highest-priority information.

In addition, our scheme uses a dynamic channel shift mechanism to communicate information about other priority levels, as shown in Figure 8. As noted above in Section 4.1, our proposal uses three communication channels and nodes share the number of channels and the order of scanning channels as well as packet priority. All nodes choose Ch1 (L1 or L2) at first. Then, if they do not detect any packets over Ch1 during ΔL_{wait} , they move to Ch2 and scan the channel again. For example, in Figure 6, nodes 6 and 7 shift from Ch1 (L1 or L2) to Ch2 (L3), and node 4 does not detect any packets over Ch2 and so shifts to Ch3 (L4), while the other nodes stay at Ch1. For this example, Table 2 summarizes the

TABLE 2: An example of SlotFrame usage in a ComFrame.

	1st SF	2nd SF	3rd SF	4th SF	5th SF	6th SF	7th SF
Root	Ch1 (L1) $R \rightarrow 1$	Ch3	Ch2 (L3) $5 \rightarrow R$	Ch3	Ch3	Ch1 (L2) $1 \rightarrow R$	Ch3
Node 1	Ch1 (L1) $R \rightarrow 1$	Ch1 (L1) $1 \rightarrow 2$	Ch3	Ch3	Ch1 (L2) $2 \rightarrow 1$	Ch1 (L2) $1 \rightarrow R$	Ch3
Node 2	Ch1 (L2) $3 \rightarrow 2$	Ch1 (L1) $1 \rightarrow 2$	Ch1 (L1) $2 \rightarrow 3$	Ch3	Ch1 (L2) $2 \rightarrow 1$	Ch3	Ch3
Node 3	Ch1 (L2) $3 \rightarrow 2$	Ch3	Ch1 (L1) $2 \rightarrow 3$	Ch1 (L1) $3 \rightarrow 4$	Ch3	Ch3	Ch3
Node 4	Ch1	Ch3	Ch3	Ch1 (L1) $3 \rightarrow 4$	Ch3	Ch3	Ch3
Node 5	Ch1	Ch2 (L3) $6 \rightarrow 5$	Ch2 (L3) $5 \rightarrow R$	Ch3	Ch3	Ch3	Ch3
Node 6	Ch2 (L3) $7 \rightarrow 6$	Ch2 (L3) $6 \rightarrow 5$	Ch3	Ch3	Ch3	Ch3	Ch3
Node 7	Ch2 (L3) $7 \rightarrow 6$	Ch3	Ch3	Ch3	Ch3	Ch3	Ch3

FIGURE 8: Outline of dynamic channel shift.

channel usage for all nodes and SlotFrames. Most TDMA-based protocols predetermine such a complete and detailed schedule as Table 2 and send it to all nodes to follow the same schedule. On the other hand, emission of a request packet from a root node to node 4 and that of a data packet from node 7 to a root node are predetermined, but other detailed slot and channel usages are autonomously and dynamically decided by our dynamic channel shift mechanism. Since control packets belonging to $L4$ use remainder of network resources, we evaluate the available bandwidth for $L4$ in Section 5.

4.5. Detailed Mechanism

4.5.1. Transmission Policy. Every node transmits a packet according to its SlotFrame usage policy and ComFrame usage policy.

4.5.2. SlotFrame Usage Policy. All nodes select a communication channel for each SlotFrame by a dynamic multichannel transmission mechanism. Then, nodes transmit $L1$, $L2$, and $L3$ packets over Ch1 or Ch2 as shown in Figure 8. They can also retransmit a packet once per SlotFrame according to the retransmission policy below. Nodes transmit $L4$ packets by CSMA/CA over Ch3.

4.5.3. ComFrame Usage Policy. A root node uses the 1st or 2nd SlotFrame to transmit an $L1$ packet to a node (final destination) in a ComFrame. It first checks the hop count to the final destination in the current network topology and the hop count of an $L1$ packet in a previous ComFrame. When the hop count of the previous $L1$ packet is even, the root node uses the 2nd SlotFrame to avoid packet collisions due to hidden terminal problems on a multihop route. A node transmits an $L2$ packet at the 1st SlotFrame when it received

FIGURE 9: Communication success probability comparison in a SlotFrame between one hop communication with retry and without retry.

an $L1$ packet in a previous *ComFrame*. Thus, an $L1$ packet and an $L2$ packet are transmitted in the same *ComFrame*. Regarding $L3$ packets, the root node notifies a node of a *ComFrame* to use for $L3$ packet transmission when a node joins the network. Each node transmits an $L3$ packet at the 1st SlotFrame of its own *ComFrame*.

Since the length of *ComFrame* is large enough for a packet sent by a node at any hop distance to reach a root node, *ComFrame* assignment can be maintained and fixed as far as the maximum hop count does not increase.

4.5.4. Retransmission Policy. The length of a time slot in a TDMA scheme like ISA100.11a is just long enough for a MAC frame of maximum size and its acknowledgement (ACK). Normally, TDMA schemes do not permit any retries in a time slot. For lossy networks, however, link quality (i.e., the communication success ratio) is significantly improved by permitting a node to send a retry packet, as shown in Figure 9. In our proposal, transmission of a retry packet is permitted for every one hop communication of $L1$, $L2$, and $L3$ packets.

Figure 10 shows a comparison of successful path transmission probabilities among the following four retransmission policies: the first policy does not support retry for either link communication or end-to-end communication; the second one supports link retry but not end-to-end retry; the third one supports end-to-end retry but not link retry; and the fourth policy supports both link and end-to-end retry. The figure shows that both retry types are effective even if the retry is only attempted once.

4.5.5. Packet Forwarding Policy. As described above for the SlotFrame usage policy, our proposed scheme does not allocate the intermediate SlotFrames of all *ComFrames*. For example, only the 3 bold SlotFrames are allocated in advance in Table 2. Every node basically seeks to forward a packet received at a previous SlotFrame with a dynamic channel shift for the transmitting process rule when it does not detect any higher-priority packets.

FIGURE 10: End-to-end path success probability.

4.5.6. Reception Process with Dynamic Channel Shift. A root node transmits an $L1$ packet to a node at the 1st or 2nd SlotFrame in each *ComFrame*. As shown in Figure 11, a nonroot node first checks Ch1 over a period of ΔL_{wait} . If it detects an $L1$ packet, it maintains the channel to receive the packet. It determines the priority level of a packet by detecting the timing. If the timing is $0 \leq \Delta L_{wait}/2$, the packet is treated as $L1$; otherwise, it is treated as $L2$. After searching Ch1, the node checks Ch2 over another period of ΔL_{wait} . If it detects a packet, it maintains the channel to receive the packet as $L3$. Otherwise, it chooses Ch3 as the communication channel at the SlotFrame.

4.5.7. Transmission Process with Dynamic Channel Shift. For transmission all nodes must check for packet existence over channels in order of priority until reaching the usage channel, as in the reception process with dynamic channel shift. The transmission process works as follows by packet priority level:

$L1$: A root node (network manager) knows the current network topology and the hop count of a node that is the destination of a previous $L1$ packet. If the hop count is even, the root node cancels transmission of an $L1$ packet at the 1st SlotFrame and reserves the 2nd SlotFrame in order to avoid the hidden terminal problem on the path.

$L2$: A node that transmits an $L2$ packet to a root node checks for packet existence over Ch1 for a period of $\Delta L_{wait}/2$. If it does not detect any packets, it transmits an $L2$ packet over Ch1.

$L3$: A node checks for packet existence over all channels in order of priority until Ch2, as in the reception process with dynamic channel shift. If the node detects no higher-priority packets, it transmits an $L3$ packet over Ch2. Otherwise, it cancels transmission of the $L3$ packet at the current SlotFrame and reserves

FIGURE 11: Outline of reception process with channel shift.

the next SlotFrame when the number of remaining SlotFrames in the ComFrame is greater than the hop counts.

L4: A node checks for packet existence over all channels in order of priority until Ch3, as in the reception process with dynamic channel shift. If the node detects no higher-priority packets, it transmits an L4 packet over Ch3 by CSMA/CA.

4.5.8. Forwarding Process with Dynamic Channel Shift. All nodes must check for packet existence, as in the transmission process with dynamic channel shift. The forwarding process works as follows by packet priority level:

L1: A node forwards a packet received at a previous SlotFrame to the next-hop node.

L2: A node checks for packet existence over Ch1 for a period of $\Delta L_{\text{wait}}/2$. If the node does not detect any packets, it forwards an L2 packet over Ch1. Otherwise, it cancels forwarding of the L2 packet at the current SlotFrame and reserves the next 3 SlotFrames.

L3: A node follows the L3 behavior in the transmission process with dynamic channel shift.

L4: A node follows the L4 behavior in the transmission process with dynamic channel shift.

5. Simulation Evaluation

5.1. Simulation Settings. To evaluate the performance impact of our priority-based dynamic multichannel transmission scheme, we performed a set of simulations with 501 nodes placed statically and randomly in a square field. A root node

was placed at the lower left-hand corner of the field, and a routing protocol for low-power and lossy networks (LLNs) [20] was applied to create routes from all nodes to the root node with a shortest-path metric. Figure 13 shows an example of the resulting network topology. This served to simulate our target system, like that shown in Figure 1. The network topology was fixed during a simulation, and a total of 10 network topologies were tested to take into consideration localization of sensors. We also assume that packet loss among neighbors is caused by several factors such as propagation models, signal processing technology, transmitting power, antenna characteristics, and reception sensitivity, except signal interference from other nodes because of TDMA-like transmission. We then determined the link PER at random as shown in Table 3. Although the link PER dynamically changes in reality, we assume it is stable and constant in this paper. Evaluation under dynamic environment is left as future work.

The network is subject to three traffic packets: request/response type traffic from a root node as unpredictable packets, sensor-to-root traffic as periodic packets, and network control traffic that exchanges information among neighbors. In our proposal, transmission of L1 and L2 packets for remote control and response is scheduled by a root node to guarantee real time communication. On the other hand, L3 packets for periodic data gathering are emitted at predetermined intervals and L4 control packets are generated irregularly. Therefore, the worst case scenario is that all of those packets are generated in a certain short period.

In this paper, we evaluate the worst case performance. More specifically, we define an AppFrame accommodating three traffic classes as shown Figure 12. An AppFrame consists of three blocks. The first block is used for a root node to send requests (L1) to all 500 nodes for remote control. Responses (L2) from nodes are also accommodated in the

TABLE 3: Simulation conditions.

Item	Notation	Value (920 MHz)	Value (2.4 GHz)
Number of nodes	N_{node}	500 nodes	500 nodes
L2 packet length	L_{data}	500 B	127 B
L2 ACK length	L_{ack}	100 B	40 B
Communication speed	v	100 kbps	250 kbps
Data collection cycle	Δ_{AF}	30 min	5 sec
Max hop count	H_{max}	8 hops	16 hops
Average hop count	H_{ave}	30 hops	8 hops
Link PER with retry	Per	0–10%	0–9%
Timeslot length	ΔL_{TS}	100 ms	10 ms
Wait time for channel shift	ΔL_{wait}	5 ms	1 ms

FIGURE 12: AppFrame composition for simulations.

FIGURE 13: Simulated network topology ($N_{\text{node}} = 500$).

same block. Block 1 is also used for periodic packets ($L3$) and control packets ($L4$). The length of Block 1 is the same as the number of nodes in ComFrames. The second block is used for retransmission of requests to those nodes from which a root node does not receive any response in Block 1. On the contrary, polling-based retransmission of periodic packets is deferred to Block 3, because periodic data gathering is more delay tolerant than remote control. A root node sends a request to resend a data packet to each node from which it fails in receiving a report in Block 1. At this time, requests and responses are given priorities $L1$ and $L2$, respectively. Control packets ($L4$) are irregularly generated in both Block 2 and 3 as in Block 1. Table 3 summarizes the details of the other parameter settings.

5.2. Simulation Results

5.2.1. End-to-End Delay of High-Priority Packets ($L1$ and $L2$).

All packets are transmitted by ComFrame. At a time when an application queues an $L1$ packet but ComFrame n is already in process, the $L1$ packet should stay in the queue until the head of the next ComFrame $n + 1$. The request is transmitted to the destination node at ComFrame $n + 1$, and the root node receives the reply packet from the destination node at ComFrame $n + 2$. Therefore, the following defines the range for the end-to-end delay time:

$$2 \times \Delta L_{\text{CF}} \leq \text{Delay}_{\text{E2E}} < 3 \times \Delta L_{\text{CF}}. \quad (1)$$

In the 920-MHz simulation case, the end-to-end delay was less than 6.6 sec ($= 3 \times 11 \text{ SlotFrames} \times 200 \text{ ms}$), while in the 2.4-GHz case, it was 1,140 ms ($= 3 \times 19 \text{ SlotFrames} \times 20 \text{ ms}$). Our proposal guarantees the deadline for remote operation, and these simulation results meet our target requirements, as listed in Table 1.

Figure 14 shows end-to-end delay comparison. We conducted field experiments to obtain delay samples of WirelessHART. In the experiments, a root node of WirelessHART received 90.2% packets (5,481 packets (received)/6,088 packets (total)) from nodes and the delay considerably fluctuates. The average delay was 1.309 sec. The theoretical maximum delay of our proposal in the similar condition is 1.14 sec and smaller than the average delay of WirelessHART. To derive this, we substitute the average link success probability of 95% in the experiment to (1).

FIGURE 14: End-to-end delay comparison.

FIGURE 15: Total number of nodes from which the root node successfully received periodic data in an AppFrame.

5.2.2. Success Rate of High-Priority Packets. Figure 15 shows the simulation results for the success rate of high-priority packets. The root node received the highest-priority packets (L1/L2) from all nodes when the link success probability (with retry in a SlotFrame) was greater than 93%. It also received all health report data (L3) from almost all nodes when the link success probability (with retry) was greater than 96%. The important points here are that our scheme guarantees the maximum delay for getting information from a node and provides a high success rate for getting packets of different priority at the same time.

5.2.3. Available Bandwidth for L4 Packets. Figure 16 shows the simulation results for the ratio of the available L4 channel usage time in which a node can totally transmit L4 packets in an AppFrame to the length of the AppFrame. According to the figure, the sum of the L4 packet bandwidth from Block 1

FIGURE 16: Total available bandwidth for L4 packets and breakdown by block.

to Block 3 was almost 90%. Although up to three instances of higher-priority traffic were generated in Block 1, the impact of the traffic was very limited. Overall, our scheme provides sufficient bandwidth, because WirelessHART requires about 30% of the bandwidth for network control packets. In addition, when the link PER becomes high, more L1, L2, and L3 packets might drop on a multihop route. This means that the total usage for L1, L2, and L3 packets would drop, and the total available bandwidth for L4 packets would increase. When the network is not stable, more network control packets should be generated in order to repair routes than when the network is stable. Therefore, this approach is appropriate for an LLN.

6. Discussion

6.1. Lower Bound of Available Bandwidth for L3 and L4 Packets. As we show above, almost all L3 packets can be delivered when the link success probability (with retry) is greater than 96%. To achieve that level, in Block 1, L1 or L2 packets and L3 packets are not often generated on the same path. ISA100.11a can allocate all time slots for all nodes in order to reduce the traffic pattern. Our proposal, however, does not schedule time slots. Instead, the root node notifies each node of a sequence number in the ComFrame at which it can transmit L3 packets. The sequence number does not depend on either the traffic pattern or network topology but may purely be in order of nodes joining the network. Therefore, we evaluated the worst case scenario in which all L1, L2, and L3 packets are transmitted on the same path or nearby paths at the same time. In this case, the end-to-end path success probability for L3 packets in Block 1 is 0% because all nodes on the path are in use. Consequently, the number of data collection packets from sensors depends on the length of Block 3 (ΔL_{Block3}).

The length of Block 3 is calculated by

$$\Delta L_{\text{Block3}} = \Delta L_{\text{AF}} - \{\Delta L_{\text{Block1}} + \Delta L_{\text{Block2}}\}, \quad (2)$$

TABLE 4: Communication patterns for $L1$ and $L2$ packets with E2E retry.

Number	Appearance pattern				Required # of ComFrames
	$L1$ (1st)	$L2$ (1st)	$L1$ (retry)	$L2$ (retry)	
(1)	Pass	Pass	—	—	1.0
(2)	Pass	Fail	Pass	Pass	2.0
(3)	Pass	Fail	Pass	Fail	2.0
(4)	Pass	Fail	Fail	—	1.5
(5)	Fail	—	Pass	Pass	1.5
(6)	Fail	—	Pass	Fail	1.5
(7)	Fail	—	Fail	—	1.0

TABLE 5: Expected number of ComFrames for $L1/L2$ packets with an E2E retry.

Index i	1	2	3	4	5	6	7
E2E success prob.	1.0	0.99	0.98	0.97	0.96	0.95	0.94
E_i	1.0	1.014	1.029	1.043	1.057	1.070	1.083

where ΔL_{Block1} is $N_{\text{node}} \times (H_{\text{max}} + 3)$, and $\Delta L_{\text{Block1}} + \Delta L_{\text{Block2}}$ is the number of devices successfully controlled with link retry and end-to-end path retry. Therefore, the length of Block 2 (ΔL_{Block2}) is calculated by

$$\Delta L_{\text{Block2}} = E_i \times N_{\text{node}} - \Delta L_{\text{Block1}}, \quad (3)$$

where E_i is the expected number of ComFrames per node for $L1/L2$ packets with both link retry and end-to-end retry. Table 4 summarizes the communication patterns for $L1/L2$ packets with end-to-end retry. The expected number of ComFrames for $L1/L2$ packets with end-to-end retry is calculated as shown in Table 5. Finally, the number of successfully received $L3$ packets (N_{L3}) is

$$N_{L3} = \Delta L_{\text{Block3}} \times \frac{1}{E_i}, \quad (4)$$

where $1/E_i$ is the success probability of round-trip end-to-end communication with end-to-end retry. In the worst case, $L3$ packets can be collected at a rate of 45.52–63.63% ($0.9 \leq 1/E_i \leq 1$).

On the other hand, $L4$ packet traffic used almost 90% of the bandwidth in our simulation scenario. In the same worst case, we assume that the communication pattern in Block 1 is similar to those in Blocks 2 and 3. Consequently, the lower bound on the available bandwidth for $L3$ and $L4$ packets (P_{L4}) is 36.36%, as calculated by

$$P_{L4} = \frac{\{(H_{\text{max}} + 3) - (H_{\text{ave}} + 3)\}}{\Delta_{L_{\text{CF}}}} = \frac{H_{\text{max}} - H_{\text{ave}}}{\Delta_{L_{\text{CF}}}}. \quad (5)$$

6.2. Compatibility with ISA100.11a Standard. Figure 17 shows how to adapt our proposal to the ISA100.11a standard protocol. Basically, our proposal is a technology between the network and data link layers, so that it does not directly affect processing in those layers. We do have to specify the operation mode and adjust some parameters of the data link layer

to compose our own frames. We use priority CAMSA/CA, select slow-hopping mode as the channel hopping pattern, and bundle time slots defined by ISA100.11a to compose SlotFrame, ComFrame, and AppFrame logically. Our scheme decides what channel (Ch1, Ch2, or Ch3) each node should use at each SlotFrame. The important point is that selecting a channel from among these three in our scheme is equivalent to deciding the operation mode in the data link layer: transmitting a packet, receiving a packet, forwarding a packet, or waiting to forward a packet. If we implement our scheme over one ISA100.11a data link function over one physical interface, the bandwidth for $L3$ and $L4$ packets will decrease. For example, in Table 2, hidden terminal problems could result. An $L4$ packet from the root node to node 5 at the 2nd SlotFrame would collide with an $L3$ packet from node 6 to node 5. Also, an $L1$ packet from node 2 to node 3 at the 3rd SlotFrame would collide with an $L4$ packet from node 4. To avoid these collisions, we can define a longer length of SlotFrame so as not to overlap the times at which all levels of packets are transmitted. Or, more specifically, our proposal implements three ISA100.11a data link functions over one physical interface in order to guarantee the maximum delay for $L1$ and $L2$ packets and keep the bandwidth for $L3$ and $L4$ packets high.

6.3. Strengths and Weaknesses of Our Proposal. Our proposal is very lightweight and much simpler than usual TDMA-based protocols like ISA100.11a. They typically have a scheduler for allocating time slots to meet application requirements. The network manager has to determine and deliver the schedule to all nodes whenever a new node joins the network or the network topology changes. It consumes considerable bandwidth and causes extra delay especially in a lossy and unstable network. Our proposal defines a ComFrame whose length is fixed during network operation. The length depends on maximum multihop count that is one of predetermined system parameters. In a ComFrame, there are at most two high primal packets (an $L1$ packet and an $L2$ packet). Moreover, the 1st SF of a ComFrame is assigned to a node to transmit an $L3$ packet. Under these setting, all nodes autonomously determine when and what packets should be transmitted on which channels to avoid packet collisions. Then, from scheduling point of view, our proposal is simple and does not need to reschedule and redeliver a

OSI layer	ISA100.11a	ISA100.11a with our proposal
Network	IPv6 (IETF 6lowpan) (i) Fragmentation (ii) Reassembling	IPv6 (IETF 6lowpan) (i) Fragmentation (ii) Reassembling
		Priority based channel shift
Data link	Time synchronized (i) CSMA/TDMA (ii) Priority CSMA (iii) Channel hopping	Time synchronized (i) CSMA/TDMA (ii) Priority CSMA (iii) Channel hopping (physical)
Physical	IEEE802.15.4- based radio	IEEE802.15.4- based radio

FIGURE 17: Comparison of functions between ISA100.11a and our proposal.

schedule even if a network topology changes. Above features are strong points of our protocols.

On the other hand, as described above in Section 6.2, our proposal requires more hardware resources than a normal ISA100.11a, when our scheme operates over ISA100.11a. Our proposal needs at least three channels to avoid collisions among different priority packets. Then a node should have three physical interfaces each of which runs full functions of ISA100.11a or have virtual communication interfaces that operate independently over a physical interface to meet our requirements. In either case, hardware cost for a node becomes more expensive than a normal of ISA100.11a. It may hinder deployment of our proposal, but IWSN should be designed to meet real time requirement to guarantee interaction within a predetermined deadline.

7. Conclusions and Future Work

This paper introduced a priority-based dynamic multichannel transmission scheme for IWSNs. Our algorithm enables transmission of packets of different priority level in the same period without collisions. The highest-priority packets for remote control can be delivered within a guaranteed deadline through a hybrid control scheme that combines centralized control by a root node and autonomous decentralized radio channel shift by nonroot nodes. At the same time, lower priority packets belonging to periodic data gathering and control can receive the satisfactory quality of service, where the collection ratio of periodic data packets is higher than 45% and the lower bound of bandwidth available to control packets is larger than 36% at the worst case scenario.

In this paper, we do not address dynamic adaptation of our scheme to handle dynamic or unexpected changes in application and system requirements. For example, composition of AppFrame must be predetermined at the deployment phase under assumptions on system configurations, but

it should be dynamically regulated to fit to actual traffic demand. In a case of an unstable network, control packets would be transmitted more frequently. Therefore, we need to organize an AppFrame to spare more bandwidth for *L4* packets. We plan to tackle these issues as future work.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] Z. Alliance, "Zigbee," 2017, <http://www.zigbee.org>.
- [2] FieldComm Group, "Hart communication," 2015, <https://fieldcommgroup.org>.
- [3] ISA, "Isa100, wireless systems for automation," 2017, <http://isa100wci.org>.
- [4] Institute of Electrical and Electronics Engineers (IEEE), "IEEE Standard for Local and metropolitan area networks-Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer IEEE Std," 802.15.4e-2012, 2012.
- [5] G. Kalani, *Industrial Process Control: Advances And Applications*, Gulf Professional Publishing, Houston, Tex, USA, 2002.
- [6] D. J. Olsen, N. Matson, M. D. Sohn et al., "Grid integration of aggregated demand response," Part 1: Load Availability Profiles and Constraints for the Western Interconnection LBNL-6417E, 2013, Lawrence Berkeley National Laboratory, <http://escholarship.org/uc/item/6ps4r3xp>.
- [7] TEPCO, "Basic concept for smart meter specification based on rfc," 2012.
- [8] D. F. Ramírez and S. Céspedes, "Routing in neighborhood area networks: a survey in the context of ami communications," *Journal of Network and Computer Applications*, vol. 55, pp. 68–80, 2015.
- [9] DLMS, "Dlms user association: Cossem identification system and interface classes," 2010.

- [10] Linear technology, "Smartmesh wirelesshart application notes," 2016, http://cds.linear.com/docs/en/application-note/Smart-Mesh_WirelessHART_Application_Notes.pdf.
- [11] Y. Li, C. S. Chen, Y. Song, and Z. Wang, "Real-time qos support in wireless sensor networks: a survey," *IFAC Proceedings Volumes*, vol. 40, no. 22, pp. 373–380, 2007.
- [12] H. Kurunathan, R. Severino, A. Koubaa, and E. Tovar, "Worst-case bound analysis for the time-critical MAC behaviors of IEEE 802.15.4e," in *Proceedings of the 2017 IEEE 13th International Workshop on Factory Communication Systems (WFCS)*, pp. 1–9, Trondheim, Norway, May 2017.
- [13] D. Vassis and G. Kormentzas, "Throughput analysis for IEEE 802.11 ad hoc networks under the hidden terminal problem," in *Proceedings of the 2006 3rd IEEE Consumer Communications and Networking Conference, CCNC 2006*, pp. 1273–1276, Las Vegas, Nev, USA, January 2006.
- [14] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.
- [15] B. Shrestha, E. Hossain, and K. W. Choi, "Distributed and centralized hybrid CSMA/CA-TDMA schemes for single-hop wireless networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 7, pp. 4050–4065, 2014.
- [16] O. D. Saputra and S. Y. Shin, "Real-time based Superframe for ISA100.11a in Wireless Industrial Network," *Journal of Communication and Computer*, vol. 12, no. 1, pp. 28–32, 2015.
- [17] I. Rhee, A. Warriier, and M. Aia, "Z-MAC: a hybrid MAC for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 511–524, 2008.
- [18] R. Costa, P. Portugal, F. Vasques, and R. Moraes, "A TDMA-based mechanism for real-time communication in IEEE 802.11e networks," in *Proceedings of the 15th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2010*, Bilbao, Spain, September 2010.
- [19] N. Q. Dinh and D.-S. Kim, "Performance evaluation of priority CSMA-CA mechanism on ISA100.11a wireless network," *Computer Standards and Interfaces*, vol. 34, no. 1, pp. 117–123, 2012.
- [20] J. Yi, T. Clausen, and Y. Igarashi, "Evaluation of routing protocol for low power and Lossy Networks: LOADng and RPL," in *Proceedings of the 2013 IEEE Conference on Wireless Sensor, ICWISE 2013*, pp. 19–24, Kuching, Malaysia, December 2013, <http://www.thomasclausen.net/wp-content/uploads/2015/12/2013-ICWISE-Evaluation-of-Routing-Protocol-for-Low-Power-and-Lossy-Networks-LOADng-and-RPL.pdf>.