A Bandwidth-Efficient MAC Scheme for Mission-Critical Applications in Industrial Wireless Sensor Networks

Hongchao Wang[1], Jian Ma[1], Youzhi Xu[2], Dong Yang[1], Hongke Zhang[1]

[1] School of Electronic and Information Engineering, Beijing Jiaotong University, China
[2] Mid Sweden University, Sweden
{hcwang, 14111026}@bjtu.edu.cn, Youzhi.Xu@ju.se, {dyang, hkzhang}@bjtu.edu.cn

Abstract

Industrial wireless sensor networks (IWSNs) have the potential to contribute significantly to a variety of wireless sensing endpoints, such as cable replacement, mobility, flexibility and cost reduction. However, the harsh and varied industrial environment entails addressing severe challenges, such as dust, heat, electromagnetic interference (EMI) and radio frequency interference from the other heterogeneous networks. One of the important challenges is the link burstiness in industrial wireless environments, which has not been heavily researched. In this paper, we propose a new timeslot scheduling algorithm to address this problem. The algorithm allows transmissions on the same link to be separated at least by a minimum timeslot distance. Moreover, to provide a synergistic complement to the timeslot scheduling, we also propose a timeslot reuse scheme, which facilitates improved efficiency and reliability in the bandwidth utilization and retransmission and decreases the average delay of the packet arrival. We evaluate the proposed algorithm and scheme by a real implementation, targeting a specific industrial application. The experimental results generally indicate that the proposed media access control (MAC) scheme greatly improves the reliability and decreases the packet arrival delay. Compared with the existing popular methods, the execution of our scheme indicates a minimal link burstiness influence and possesses more efficient bandwidth utilization.

Keywords: Industrial wireless sensor networks, Multi-hop wireless network, Link burstiness, Real-time communication, Retransmission efficiency

1 Introduction

Traditionally, industrial automation systems are realized through wired communications. However, wired automation systems require installation and regularly scheduled maintenance of expensive communication cables; thus, they are not widely implemented in industrial plants because of their high cost [1]. Therefore, the wireless sensor network (WSN) was developed as a network of microelectromechanical systems, and it is more prevalent in process automation applications [2]. The features of the WSN provide several advantages over the traditional wired industrial monitoring and control systems, including flexibility, rapid deployment, self-organization and intelligent-processing capability. As a key technology in the recent surge of research in the areas of Industry 4.0 and Factories of the Future (FoF), industrial wireless sensor networks (IWSNs) can be extended to include applications in industry [3-4]. However, the harsh and varied industrial environments and the demanding requirements also admit severe challenges for IWSNs, such as radio frequency interference, highly reliable and timely data requirements and additional industrial expertise. Therefore, due to differences from traditional WSNs in terms of the vastness of its domain, IWSNs have been divided into classes that depend upon functional and service requirements. Moreover, the increase of spectrum utilization in non-licensed bands along with the reduced power used by these nodes is expected to cause high interference problems in IWSNs. Therefore, this paper not only focuses on the general challenges but also on a specific problem in IWSNs, which is the problem of link burstiness.

Several types of standards exist for IWSNs, most importantly, ZigBee [5], ISA100.11a [6], WIA-PA [7] and WirelessHART [8]. However, these standards are designed as customizable frameworks that meet the requirements of a particular application setting. Recently, most of the research in IWSNs focused primarily on the network performance related to reliable scheduling [9-10], criticality [11-12], mobility [13-14], routing protocol [15-16], end-to-end delay analysis [17-19] and energy efficiency [20-22]. WirelessHART is one of the most popular standards designed to meet the requirements for process automation systems. Several approaches, such as channel hopping and the redundant routing path, are used to provide high end-to-end reliability. However,

*Corresponding Author: Hongchao Wang; E-mail: hcwang@bjtu.edu.cn
DOI: 10.3966/160792642018051903015

these approaches do not address packet loss because of the link burstiness. The link burstiness is a physical property, which means that transmissions on a wireless link do not have an independent failure probability; instead, they have periods of continuous message loss. However, few major studies address the problem of the link burstiness, which can greatly deteriorate the performance of the network system. However, many studies substantiate that the link burstiness can greatly affect the network performance, especially in industrial environments [23-24]. Moreover, even on similar networks, the network communication performance may be different from each other due to the link burstiness.

Few studies in the literature address the problem of the link burstiness. In [25], the paper proposes two metrics of expected future transmissions (EFT) and MAC3 for the estimation of the link burstiness and introduces a link burstiness estimator. However, the link burstiness estimation placed a large requirement on statistical measurements and could not match the speed of the channel variation. In [9], the authors propose a novel method based on the segmented slot assignment, fast slot competition and free node concept, which can improve the reliability and real-time communication. However, the methods do not consider the problem of the link burstiness and the segmented timeslots may increase the end-to-end packet delay. In addition, [26] proposes a novel link scheduling scheme using a virtual token to identify the utilization of the shared links. Although this scheme can allow a more flexible use of the available links at each point in time, the increased energy consumption cannot be ignored, since all of the nodes in a path remain in a reception mode before correctly receiving the message. In [23], the authors study the statistical and temporal properties of the links and develop some new routing algorithms that consider consecutive same path link lagged correlations. In addition, Srinivasan K. defines a metric, β, to measure the link burstiness in [24], using it to make the protocol pause after encountering a packet failure to reduce its transmission cost. In [27], Munir S. also defines a new metric, Bmax, that characterizes the links by their maximum burst length; the solution is to choose a novel least-burst-route, which minimizes the sum of worst case burst lengths over all links in the route. In another approach [28], the authors describe a burst-link-aware routing protocol achieved by exploiting an efficient burstiness identification method (BIM) that passively measures the RSSI value of packets received from neighbors. However, given the time requirement for the link classification, their approaches were unable to satisfy the diversified applications in varied industrial environments.

The main motivations of this paper are to address the challenges, including slot resource constraints, inefficient shared slot competition and, most importantly, the link burstiness. To overcome these challenges, this article presents a novel slot scheduling method supporting the minimum timeslot distance between two transmissions on the same link for a TDMA-based multi-hop tree wireless network. The proposed solutions are general for most of the industrial applications, and the main contributions of this paper can be summarized as follows:

(1) The situation of link burstiness is unpredictable, which may cause a packet loss, especially with the occurrence of a consecutive burst link. Thus, the proposed timeslot scheduling algorithm takes this problem into consideration and allows the transmissions on the same link to be separated by at least a minimum timeslot distance. To the best of our knowledge, this study is the first to address the problem of the link burstiness in the consideration of timeslot scheduling.

(2) Due to the unavoidable challenges of resource constraints, we propose a timeslot reuse scheme to reduce the waste of timeslots caused by a packet loss. The proposed scheme works well together with the proposed timeslot scheduling algorithm and makes the bandwidth utilization more efficient.

(3) The harsh and varied industrial environment can make currently issued MAC protocols unpractical. Thus, the proposed algorithm and scheme are practically implemented on a real wireless sensor network for an industrial application. The results generally indicate that the proposed MAC scheme achieves greater bandwidth efficiency and more reliability.

The rest of this paper is structured as follows. Section 2 introduces the system model and defines the symbols for the preliminary proposed scheduling. Section 3 describes our proposed timeslot scheduling algorithm and the reuse scheme. We report the results of the practical experiments in Section 4. The study's conclusion is presented in Section 5.

2 System Model

An IWSN consists of a collection of sensor nodes, actuators, a gateway and a network manager, as shown in Figure 1. In this paper, we focus on the monitoring and control traffic in our system. Each sensor node periodically collects the sampling data from the corresponding industrial device and sends them to the gateway through a wireless network. A node is not only a packet generator but also a packet relay, which can form a multi-hop wireless network. The gateway has the ability to relay the packets from the sensor nodes to the network manager. Because we adopt centralized management, the network manager is responsible for managing the entire network. Therefore, the communication resources schedule is issued to all of the nodes by the network manager after the network joining process.

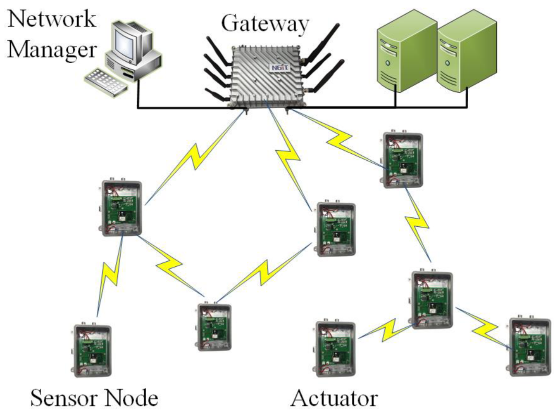


Figure 1. A system of an IWSN

We first assume that a sensor network is a static graph G = (ℕ, 𝕋), where ℕ is the set of all nodes in the network and 𝕋 is the routing tree rooted at the gateway. We denote the set of all nodes as ℕ = $\{N_\gamma \mid \gamma = 1, 2, \ldots, \Gamma\}$, where Γ is the total number of nodes and the gateway is N_0. We also assume that the transmission deadline of each sensor node is equal to its update rate. To better illustrate the proposed algorithm, we distributed the data communication tree in different generations $l \in \{1, 2,\ldots, L\}$ according to the depth of the node in the tree, where L is the depth of 𝕋. The gateway N_0 has I direct sons, and each of them constructs a subtree $T_i \in$ 𝕋, $i \in \{1, 2,\ldots, \mathrm{I}\}$ that is rooted at the corresponding node. The sequence of subtrees is ordered according to three parameters with a descendent priority degree: (*a*) The total number of nodes I_i in the subtree T_i and the largest first; (*b*) The total number of generations L_i in the subtree T_i and the largest first; (*c*) The original node number γ and the smallest first. Accordingly, the node order of each subtree follows the new number system. In general, any node in 𝕋 can be denoted as $N_{i,l,k}$, where k is the node number in the l-th generation of T_i, $k \in \{1, 2, \ldots, \Gamma_{i,l}\}$, and $\Gamma_{i,l}$ is the total number of nodes in the l-th generation in T_i. For clarity, we list the symbols used in this paper in Table 1. If $l = 1$, it means there is only one node in the subtree, the root of T_i. If $2 \leq l \leq L$, we order the node number k according to the principle: if $\gamma < \gamma'$, then $k < k'$. For example, there is a one-to-one mapping relationship of the nodes, as shown in Figure 2, i.e., $(N_{1,1,1}, N_{2,1,1})=(N_6, N_4)$, which are the roots of the set of the subtrees. We also have, for example, $N_{1,3,2}=N_5$ and $N_{2,2,2}=N_3$. A subtree is represented as a matrix $T_i =(\{N_{i,l,k} / N_\gamma, \{\mathrm{RT}_{i,\mathrm{I},k}\})$, where $\mathrm{RT}_{i,\mathrm{I},k}$ is the route from the sensor node $N_{i,l,k}$ to the gateway N_0 or from N_0 to the actuator $N_{i,l,k}$. For simplicity, we assume all of the nodes are sensors, and they have the same update rate for periodic transmissions.

Table 1. Symbols used in our scheme

Symbol	Description
T_i	the i-th subtree rooted at a node in the first generation (l=1)
l	generation number in 𝕋, the gateway is the 0-th generation.
L_i	the maximum number of generations in T_i, $1 \leq L_i \leq \mathrm{L}$
I	the number of subtrees
Γ	the total number of nodes in the tree 𝕋
Γ_i	the number of nodes in subtree T_i
$\Gamma_{i,l}$	the number of nodes in the l-th generation of the subtree T_i
$N_{i,l,k}$	A node in the l-th generation of the subtree T_i

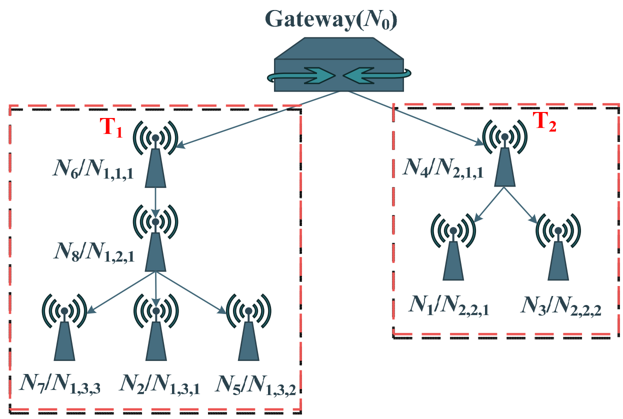


Figure 2. An example of a mapping tree

3 Segment Schedule

In this section, we present the minimum link reuse distance and the proposed algorithm for scheduling the timeslot table in a periodic superframe. Meanwhile, we also utilize a scheme of timeslot preemption to reuse the wasted timeslots for improving the bandwidth efficiency.

3.1 Minimum Link Reuse Distance

Suppose all of the nodes periodically generate a data packet with the same update rate, and all transmissions are allocated to corresponding slots. In a routing tree, a node in the l-th generation requires l dedicated slots for its packet transmission. Therefore, the nodes in the entire tree require a total of J' dedicated timeslots for packet transmissions in a periodic superframe, where J' can be calculated by (1).

$$J' = \sum_{i=1}^{\mathrm{I}} \sum_{l=1}^{L_i} l \cdot \Gamma_{i,l} \qquad (1)$$

The repeating period time of a superframe is equally divided into the timeslots and the whole superframe is divided into three parts. The first part consists of J timeslots ($J \geq J'$), which are assigned to dedicated transmissions and several distributed shared timeslots for retransmissions; the entire second part is assigned to the shared timeslots, which are used for retransmissions; the timeslots in the remaining part are reserved for network management and other purposes. In this paper, the timeslots in the first part comprise the dedicated schedule table, and we use $\mathbb{S} = \{s_j \mid j = 0, 1,\ldots, J-1\}$ to denote it. The heavier traffic always occurs at the nodes which are obviously in the first generation of a tree, and the heaviest traffic is possessed by the node which has the maximum number of nodes Λ in its subtree, where Λ is

$$\Lambda = \max\left\{\Gamma_i \mid i = 1, 2, \ldots, \mathrm{I}\right\} \quad \textbf{(2)}$$

To reduce the influence of intermittent link failure, we consider that the transmissions on the same link are not allocated to the consecutive timeslots or the timeslots are not close. Thus, we define the minimum link reuse distance $D_{\min}$, where the minimum number of timeslots between two consecutive transmissions on the same link is $D_{\min}-1$. Theoretically,

$$D_{\min} = \left\lceil \frac{J'}{\Lambda} \right\rceil + \tau \quad \textbf{(3)}$$

where $\lceil x \rceil$ is the minimum integer such that $\lceil x \rceil \geq x$ and τ ($0 \leq \tau \leq \tau_{\max}$) is the average number of shared timeslots in each subpart. The value of τ is dependent on the length requirement of the minimum distance and generally set at 1. This implies that, if the length of a burst noise or interference is less that the minimum link reuse distance in any link, at most one packet on the link is lost due to the corruption, and the link recovers after the interval of the minimum link reuse distance.

We divided the J timeslots into a set of subparts $\{S_\lambda \mid \lambda = 0, 1,\ldots, \Lambda-1\}$, and the length of each segment is equal to $D_{\min}$, where J is

$$J = \Lambda \cdot D_{\min} \quad \textbf{(4)}$$

The reasons for adding $J-J'$ shared timeslots are (a) to increase the minimum link reuse distance; (b) to reduce the collision degree in the shared timeslots in comparison to all of the shared timeslots in the second part of superframe; (c) to ease the dedicated scheduling. The bandwidth of a resource-constrained IWSN is limited, i.e., the number of timeslots in a superframe is constrained by the period, and the length of the shared timeslots in the second part must be equal to or greater than the minimum link reuse distance. Thus, the value of $D_{\min}$ is restricted.

3.2 Algorithm for Scheduling the First Part of the Superframe

To guarantee that the timeslot interval between the two transmissions on the same link is equal to the minimum link reuse distance, we designed a timeslot scheduling algorithm shown in Algorithm 1. The algorithm focuses only on the first part of the superframe and assigns the corresponding dedicated timeslot to each transmission according to the tree, which we have renumbered. First, an empty timeslot table with the length of $J-1$ timeslots is constructed, which constitutes the first part of the superframe. We traverse the entire route $\mathrm{RT}_{i,I,k}$ that belongs to the nodes in each subtree. According to the principle of subtree generation, T_1 has the maximum generations and links, thus the nodes which locate in a larger number of hops from the gateway need a larger number of continuous timeslots to transmit a packet to the gateway. Therefore, the sequence of the numbered nodes is derived from the bottom of each subtree, and the nodes with smaller k have a higher priority to be traversed when they are in the same generation and in the same subtree (lines 4, 8 and 9). This step will mitigate the degree of collisions, meaning that one timeslot is assigned to two or more nodes to transmit.

Algorithm 1

```
1:  Input: T = {T_i : {Γ_i, {N_{i,l,k}/N_γ, {RT_{i,I,k}}}} | i = 1, 2,…, I}
2:  Output: 𝕊 = {(s_j, N_γ → N_γ) | j = 0, 1, …, J − 1}
3:  Initiate: part I, part II, j′ ;
4:  for each T_i ∈ 𝕋
5:    if Γ_i ≥ 3 then
6:      j_a = (i−1)D_min mod (J−1) + L_i−1;
7:    else  j_a = L_i−1 ;
8:    j_b = J−1 ;
9:    for l = L_i to 1 do
10:     for k=1 to Γ_{i,l} do
11:       part I = False ;
12:       part II = True ;
13:       if j_a > J − 1 then
14:         j_a = l − 1;
15:         j_b = j′ ;
16:       for j = j_a to j_b do
17:         if s_{j-l+1},…, s_j = φ then
18:           Assign [s_{j-l+1}, s_j] to (N_{i,l,k}/N_γ, RT_{i,I,k});
19:           if N_{i,l,k} = N_{i,Li,1} then j′ = j − D_min ;
20:           j_a = j + D_min ;
21:           part I = True ;
22:           part II = True;
23:           break ;
24:       if part I = False and part II = True then
25:         j_a = l − 1;
26:         j_b = j′;
27:         part II = False;
28:         go to line 16 ;
29:       else if part II = False then
30:         return UNSCH;
31: return SCH, 𝕊 ;
```

After locating the node ready to be scheduled, we need to search the timeslot scheduling table and find enough unscheduled consecutive timeslots for the transmissions in the corresponding $RT_{i,l,k}$. To avoid conflict and decrease the complexity of the algorithm, the start of the timeslot table search is derived from the different timeslots according to the sequence of the subtree and the number of nodes in the subtree (lines 6-8). We define several temporary variables, which are used for searching the available unscheduled timeslots. *Part I* represents whether the timeslot allocation to a node's routing transmission succeeds in the first round of the search. Similarly, if *part II* changes to false, there are not enough timeslots to be assigned in the second round of the search. Each round of the search ranges from the timeslot j_a to j_b, which depends on the following situations. When it begins mapping a new subtree T_i, j_a is initialized to i-th $D_{\min}$ timeslot away from L_i-1 if the subtree contains at least three nodes, otherwise, $j_a=L_i-1$. In each loop of searching the same subtree, j_a is set to be a minimum distance away from the last assigned timeslot in the first round of the search. In the above situations, $j_b=J$-1. In addition, $j_a=l-1$ when it has failed to find enough unscheduled timeslots in the first round of the search or when the beginning of searching for the timeslot is beyond the end of the dedicated timeslot. The j_b timeslot is set to be the last assigned timeslot of the first node in a subtree, which is recorded by a temporary variable j'. During the process of scheduling, the algorithm will return the unschedulable state (UNSCH) if there are no continuous timeslots for all transmissions of any $RT_{i,l,k}$, otherwise, the algorithm will return a scheduled timeslot table $\mathbb{S}$ and schedulable state (SCH).

In Algorithm 1, the **for** loops in lines 4, 8, 9 are used to search each node in the whole tree. Thus, the time complexity of the three **for** loops is O(Γ). Since we need to find enough unscheduled timeslots in the superframe before allocating to a node, the worst-case is searching the entire superframe. Therefore, the time complexity of the algorithm is $O(\Gamma \cdot J)$.

3.3 Timeslot Reuse Scheme

To achieve efficient wireless bandwidth utilization, we assign l-hops transmissions in l consecutive timeslots. Therefore, the timeslot reuse scheme is proposed to avoid wasting the timeslots, which occurs because of the packet loss of the previous transmission in a packet flow. Every node executes the timeslot reuse scheme at each timeslot and finally updates the state of the radio. Algorithm 1 is executed on the network manager and the released timeslot table is stored by each node after joining the network. We use $[S_\alpha, S_\beta]$ to denote the range of timeslots, which belongs to a packet flow from the sensor node to the gateway or from the gateway to the actuator. Each node extracts its own timeslot ranges according to the released timeslot table, and every node may have several ranges within a superframe, since a node may belong to different packet flows. Because the timeslot table is fixed in a node, it may cause the timeslots to be wasted. Therefore, the timeslot reuse scheme is proposed to improve the efficiency of the timeslot utilization, as shown in Algorithm 2. In each timeslot, a node has three radio states, i.e., *transmitting*, *receiving*, *sleeping* and *sharing*, where the first three states correspond to the corresponding states of the radio transceiver, and the *sharing* state represents that the node executes the slotted CSMA/CA mechanism at the current timeslot. We employ the mechanism proposed in [29] to improve the reliability of the shared timeslots competition.

Algorithm 2. Timeslot reuse scheme

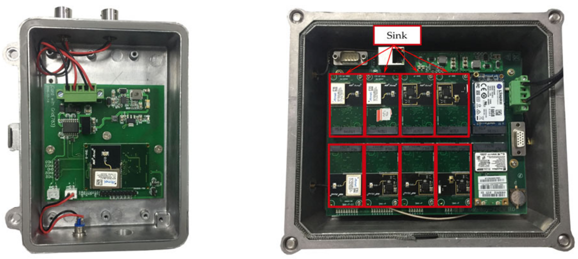
```
Input: Current timeslot s and the Dedicated_state in s
Output: Current_state
if  Dedicated_state = Transmitting  then
  if  pkt_c exists in Q  then
    Current_state = Transmitting
  else  Current_state = Receiving
else if  Dedicated_state = Sleeping  then
  if  S_α ≤ s ≤ S_β  &  pkt_c exists in Q  then
    Current_state = Transmitting
  else  Current_state = Sleeping
else  Dedicated_state = Sharing   then
  if  Q ≠ ϕ    then
    Current_state = Sharing
  else  Current_state = Receiving
else  Current_state = Receiving
if   s = S_β   then
  Update S_α and S_β  to next active timeslot range
  if  pkt_c exists in Q   then
    Change pkt_c to pkt_r
return Current_state
```

When a new timeslot s is upcoming, a node needs to update its radio state (*current_state*) according to certain judgements, which are (a) the original state (*dedicated_state*) in the timeslot table; (b) whether the packet that belongs to the current flow (pkt_c) exists in the packet queue ($\boldsymbol{Q}$); (c) whether the current timeslot is within the latest timeslot range of the current packet flow. If a packet loss occurs, and the packet that was supposed to be transmitted on the dedicated timeslot has not arrived at the gateway, this packet (pkt_c) will be marked as pkt_r, which needs to be retransmitted.

4 Evaluation

4.1 Description of the Experiments

To verify the performance of the proposed MAC scheme, the experiments were conducted in a practical application. We used an LPC1769 mote with an AT86rf212B radio transceiver as the sensor nodes. The access point, represented by sink, was embedded on the gateway and communicated with the sensor nodes and the gateway by a wireless and USB port, respectively. The IWSN manager was a generic Intel x86 server, and the gateway was a generic IPC using an Intel Atom processor. The expansion board used to connect multiple APs in the gateway was designed by the authors. The hardware of the sensor node and the gateway are shown in Figure 3(a) and Figure 3(b), respectively. The details of the remaining parameters are listed in Table 2. The experiments were performed in an 80*120 m^2 industrial factory, and each node was required to periodically collect the information about the electricity by an RS485 interface.



(a) Node (b) Gateway

Figure 3. The hardware of the sensor node and the gateway

Table 2. The parameters of the sensor node

Parameters	Value
Microcontroller	Cortex-M3
Power of Transmission	10 dBm
Power of Interference Transmission	-3 dBm
Rate of Transmission	250 kb/s
Length of Superframe	1 s
Size of Packet	80 Bytes
Frequency	780 MHz

To purposely generate the link burstiness, some interference nodes were placed around each sensor node and the gateway. The transmission power of the interference nodes was −3 dBm, since the corresponding interference node can only interfere with the specific link and cannot influence link transmissions. We adjusted the length of the link burstiness by changing the number of interference packets sent by the interference nodes. We randomly generated six interference timeslot sequences as the link burstiness. All of the interference nodes were also synchronized with the network, and the transmission of the interference packets followed the TDMA format. Each experiment was sustained for 10 minutes, which contained 60,000 timeslots. We randomly selected the NL (number of links) timeslots among these 60,000 timeslots, and the links allocated to the selected timeslots would then encounter the link burstiness caused by the interference nodes, where the length of the link burstiness is BL. Since we controlled the distance between the interference node and sensor node, other links were not influenced by the link burstiness. The corresponding BL and NL in each experiment are listed in Table 3.

Table 3. The BL and NL in each experiment

BL (timeslots)	5	10	15	20	25	30
NL	600	500	500	400	400	400

4.2 Evaluation and results

In the evaluation, we used three alternative schemes for comparisons. One scheme applied the segment schedule proposed in [9]. This scheme constructs the timeslot table by changing the level of the nodes in the routing tree. The second scheme used a random flow-based schedule, where the scheme constructs the timeslot table in the sequence of randomly selecting the flows, and the consecutive timeslots are allocated to each flow. Specifically, we added the timeslot reuse scheme that we proposed to the second scheme. The last scheme used our proposed algorithm that worked together with the timeslot reuse scheme. As shown in Figure 4, we used the same tree topology in all experiments for accurate comparisons. The topology corresponded to our industrial application, which consisted of 26 sensor nodes. According to our system model, these 26 sensor nodes constituted eight subtrees ordered by the corresponding priorities (T_1-T_8). For clarity, we display the three timeslot tables scheduled by the three schemes in Figure 5(a), Figure 5 (b) and Figure 5(c). The shared timeslots are inserted behind each segment of the dedicated timeslots in the segment schedule. The assignment was to put the shared timeslots behind all of the dedicated timeslots in both the random flow-based schedule and our scheme. Additionally, there are still several distributed timeslots inserted into the dedicated timeslots according to our algorithm. However, we maintained the same number of shared timeslots in each scheme for each comparative experiment.

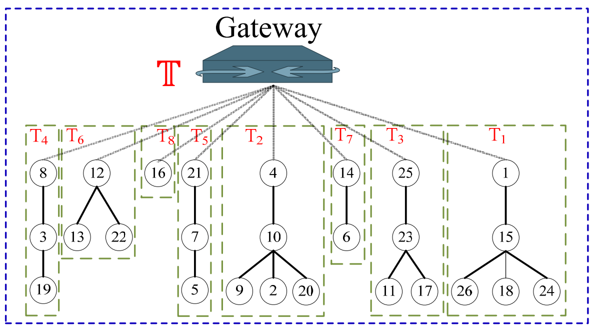


Figure 4. The tree topology of experiments

	Segment #1															Segment #2								Segment #3						
Quantity	1	1	1	1	1	1	1	1	1	1	1	1	1	1	x1	4	4	3	2	2	3	2	x2	5	5	4	3	3	x3	Reserved
Link	18 → 15	24 → 15	26 → 15	2 → 10	9 → 10	20 → 10	11 → 23	17 → 23	19 → 3	5 → 7	13 → 12	22 → 12	6 → 14	16 → G	x → x	15 → 1	10 → 4	23 → 25	3 → 8	7 → 21	12 → G	14 → G	x → x	1 → G	4 → G	25 → G	8 → G	21 → G	x → x	

(a) The timeslot table scheduled by the Segment schedule

Slot No.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Link	23 → 25	25 → G	3 → 8	8 → G	21 → G	11 → 23	23 → 25	25 → G	13 → 12	12 → G	14 → G	1 → G	5 → 7	7 → 21	21 → G	12 → G	6 → 14	14 → G	2 → 10	10 → 4	4 → G	10 → 4	4 → G	16 → G	19 → 3	3 → 8	8 → G	4 → G	20 → 10	10 → 4	4 → G

Slot No.	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54 ę 99
Link	22 → 12	12 → G	8 → G	9 → 10	10 → 4	4 → G	24 → 15	15 → 1	1 → G	7 → 21	21 → G	17 → 23	23 → 25	25 → G	26 → 15	15 → 1	1 → G	25 → G	15 → 1	1 → G	18 → 15	15 → 1	1 → G	Shared and Reserved

(b) The timeslot table scheduled by the random flow-based schedule.

	S_0												S_1												S_2						
Slot No.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Link	18 → 15	15 → 1	1 → G	4 → G	25 → G	8 → G	5 → 7	7 → 21	21 → G	13 → 12	12 → G	16 → G	24 → 15	15 → 1	1 → G	2 → 10	10 → 4	4 → G	x → x	7 → 21	21 → G	22 → 12	12 → G	x → x	26 → 15	15 → 1	1 → G	9 → 10	10 → 4	4 → G	11 → 23

						S_3												S_4												
Slot No.	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60 ę 99
Link	23 → 25	25 → G	21 → G	12 → G	6 → 14	14 → G	15 → 1	1 → G	20 → 10	10 → 4	4 → G	17 → 23	23 → 25	25 → G	19 → 3	3 → 8	8 → G	14 → G	x → x	1 → G	x → x	10 → 4	4 → G	x → x	23 → 25	25 → G	x → x	3 → 8	8 → G	Shared and Reserved

x → x (blue) : Distributed shared timeslots in our scheme; x → x (green) : Segment shared timeslots in Segment schedule

(c) The timeslot table scheduled by our proposed algorithm.

Figure 5. The three timeslot tables scheduled by the Segment schedule, random flow-based schedule and our scheme

Two metrics were measured to examine the performance of our scheme, which are the Packet Arrival Ratio (PAR) and the average end-to-end delay. The PAR represents the successful transmission ratio of all the flows by direct transmission or retransmission in the current superframe since the failed packets will be abandoned at the end of current superframe. We changed the number of shared timeslots to observe the PAR variance of each scheme. As shown in Figure 6, the PAR of the three schemes increases along with the increase of the shared timeslots. It is obvious that the PAR of our scheme is better than the other two schemes. We can see that the performance of the segment schedule deteriorates as the burstiness lengthened. The reasons for the diminished performance are that the burstiness can interfere with a link for several consecutive timeslots, and the segment is separated according to the node level. If the link of a leaf node undergoes interference due to the burstiness, then the packet fails to be transmitted and retransmitted in the current segment, and the timeslots related to that packet in the next segment will be wasted. This situation requires more shared timeslots to maintain reliability, but it causes the bandwidth utilization to be inefficient. However, our scheme has minimal influence from the BL increase. This occurs because we divided two of the same-link transmissions by the minimum timeslot distance and used the timeslot reuse scheme to make bandwidth utilization more efficient. When the BL reaches 30, 17 shared timeslots can guarantee a high PAR. However, the segment schedule needs 26 shared timeslots. The trend of the PAR is similar in the random flow-based schedule and our scheme. This occurs because we added the timeslot reuse scheme into the random flow-based schedule, and our scheme is also a flow-based schedule.

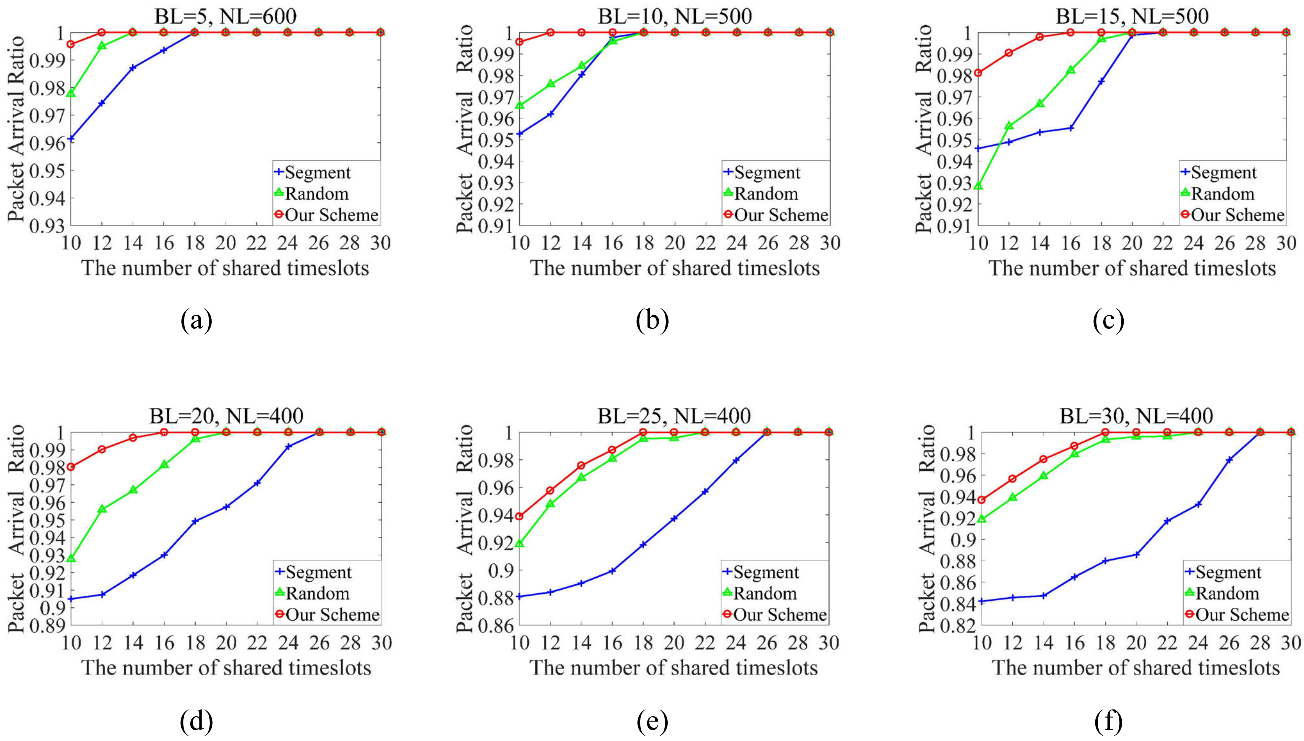


Figure 6. The PAR of three schemes with different BL and NL

The average packet end-to-end delay of the three schemes is displayed in Figure 7. We measured the packet delay separately, which belongs to the flows with different lengths. The depth of the tree is three, thus we distinguished three types of flow, i.e., one-hop flow, two-hop flow and three-hop flow. We measured the time of packet generation at one timeslot before the first transmission of the packet flow. The average packet end-to-end delay for one-, two- and three-hop flow is shown in Figure 7(a-c), Figure 7(d-f) and Figure 7(g-i), respectively. In addition, 10, 20 and 30 shared timeslots were selected in each delay measurement. The results of one-hop flow measurements show that the average end-to-end delay of the segment schedule was generally lower than the flow-based schedule. As the BL increased, the average end-to-end delay of the segment schedule increased, and it approximated the delay of the flow-based schedule when the BL reached 20. However, the delay of the two-hop and three-hop flows clearly reflected the advantage of our scheme and the random flow-based schedule. These results indicate that the gap of the delay between the segment schedule and our scheme is approximately 250ms-630ms. The reason for such a large delay gap is that if the packets failed to be successfully transmitted in the previous segment, they are only transmitted at the shared timeslots of the next segment. In addition, as the shared timeslots increased, the time interval between the transmissions of the different hops also increases. Moreover, the delays of the random flow-based schedule and our scheme are minimally influenced by the length of the burstiness and the number of shared timeslots. The timeslot allocation of the transmissions on the same link is dispersed, especially in our scheme, which is dispersed by a minimum distance. In summary, the performance of our scheme is illustrated by its reliability, timeliness and bandwidth efficiency.

5 Conclusion

In this paper, we consider the problem of link burstiness to address timeslot scheduling. Thus, we proposed a novel timeslot scheduling method that supports the minimum timeslot distance between two transmissions on the same link for TDMA-based multi-hop tree wireless networks. We also proposed a timeslot reuse scheme to reduce the waste of timeslots caused by a packet loss. The proposed scheme works synergistically with the proposed timeslot scheduling algorithm and makes the bandwidth utilization more efficient. In addition, our timeslot scheduling algorithm and scheme are implemented in a real industrial application. The results obtained in the experiment indicate that our proposed methods significantly reduced the undesirable influence of the link burstiness and improved the reliability. Compared with existing timeslot scheduling algorithms, the timeslot reuse scheme can greatly improve the efficiency of the bandwidth utilization. Moreover, the proposed solutions are sufficiently general for most industrial applications.

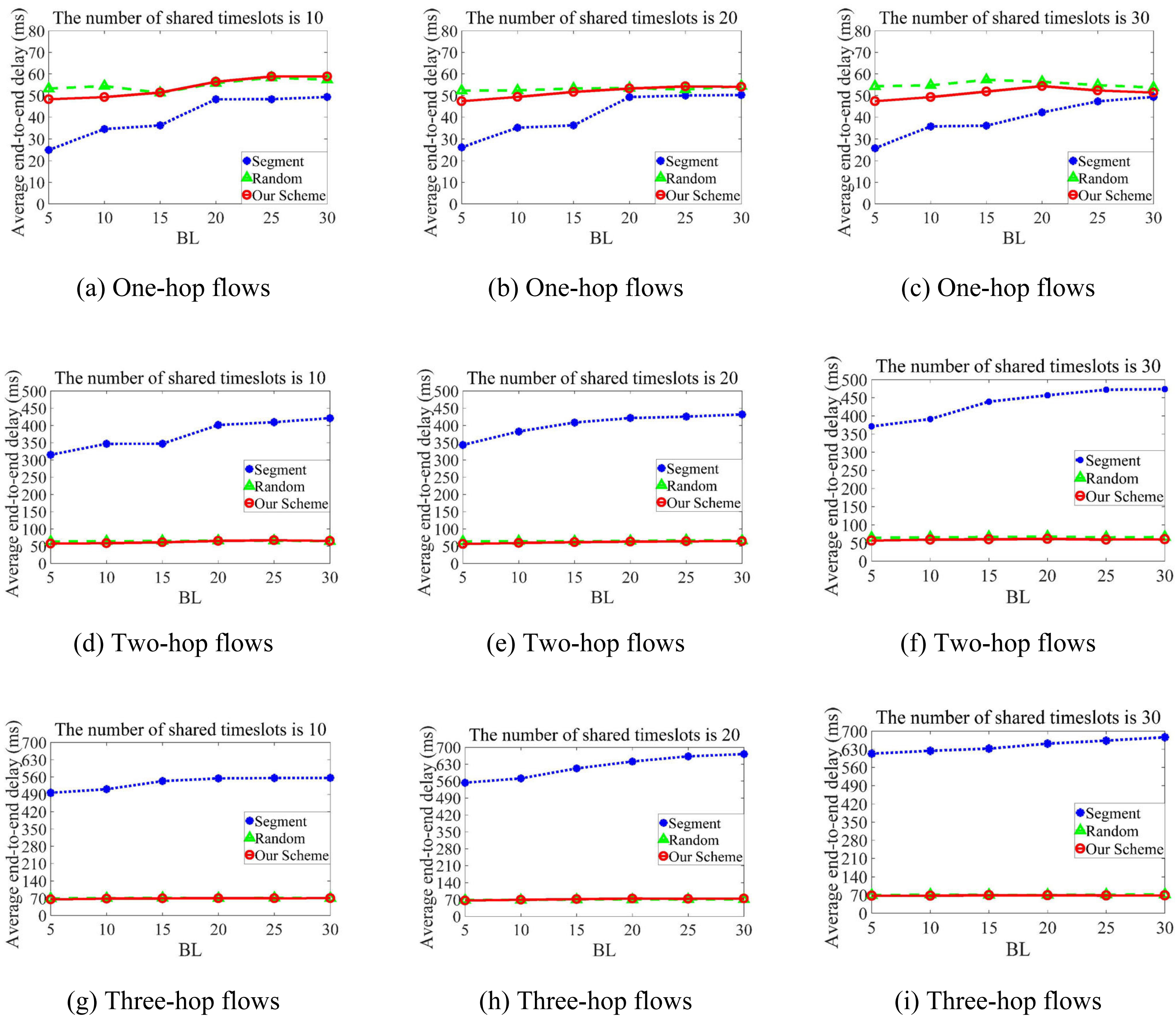


Figure 7. The average end-to-end packet delay comparison of the three scheme on different types of flows

Acknowledgements

This work was supported by the Fundamental Research Funds for the Central Universities under Grant No. 2016JBM004, and supported by Project 61701018 of National Natural Science Foundation of China.

References

[1] V. C. Gungor, G. P. Hancke, Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches, *IEEE Transactions on Industrial Electronics*, Vol. 56, No. 10, pp. 4258-4265, October, 2009.

[2] A. A. K. Somappa, S, K. Ovsthus, L. M. Kristensen, An Industrial Perspective on Wireless Sensor Networks: A Survey of Requirements, Protocols, and Challenges, *IEEE Communications Surveys & Tutorials,* Vol. 16, No. 3, pp. 1391-1412, November, 2014.

[3] X. Li, D. Li, J. Wan, A. V. Vasilakos, C. F. Lai, S. Wang, A Review of Industrial Wireless Networks in the Context of Industry 4.0, *Wireless Networks*, Vol. 23, No. 1, pp. 23-41, January, 2017.

[4] European Commission, *Factories of the Future*, http://www.effra.eu/factories-future-roadmap.

[5] Zigbee Alliance, *Zigbee Specification*, http://www.zigbee.org/download/standards-zigbee-specification.

[6] International Society of Automation, *Wireless Systems for Industrial Automation: Process Control and Related Applications*, 2011.

[7] International Electrotechnical Commission, *IEC 62601 Ed.1: Industrial Communication Networks- Fieldbus Specifications-WIA-PA Communication Network and Communication Profile*, 2011.

[8] D. Chen, M. Nixon, A. Mok, *WirelessHART: Real-Time Mesh Network for Industrial Automation*, Springer, 2010.

[9] D. Yang, Y. Xu, H. Wang, T. Zheng, H. Zhang, H. Zhang, M. Gidlund, Assignment of Segmented Slots Enabling Reliable Real-Time Transmission in Industrial Wireless Sensor Networks, *IEEE Transactions on Industrial Electronics*, Vol. 62,

No. 6, pp. 3966-3977, June, 2015.
[10] S. Han, X. Zhu, A. K. Mok, D. Chen, M. Nixon, Reliable and Real-Time Communication in Industrial Wireless Mesh Networks, *IEEE Real-Time and Embedded Technology and Applications Symposium*, Chicago, IL, 2011, pp. 3-12.
[11] C. Xia, X. Jin, L. Kong, P. Zeng, Bounding the Demand of Mixed-Criticality Industrial Wireless Sensor Networks, *IEEE Access*, Vol. 5, pp. 7505-7516, January, 2017.
[12] X. Jin, C. Xia, H. Xu, J. Wang, P. Zeng, Mixed Criticality Scheduling for Industrial Wireless Sensor Networks, *Sensors*, Vol. 16, No. 9, pp. 1376, September, 2016.
[13] J. Ma, D. Yang, H. Zhang, M. Gidlund, A Reliable Handoff Mechanism for Mobile Industrial Wireless Sensor Networks, *Sensors*, Vol. 17, No. 8, p. 1797, August, 2017.
[14] H. Fotouhi, M. Zuniga, M. Alves, A. Koubaa, P. Marron, Smart-HOP: A Reliable Handoff Mechanism for Mobile Wireless Sensor Networks, *European Conference on Wireless Sensor Networks*, Trento, Italy, 2012, pp. 131-146.
[15] V. Sharma, K. Srinivasan, QoS-Aware Routing in Wireless Networks Using Aerial Vehicles, *Journal of Internet Technology*, Vol. 19, No. 1, pp. 73-89, January, 2018.
[16] L. Wu, L. Nie, B. Liu, J. Cui, N. Xiong, An Energy-balanced Cluster Head Selection Method for Clustering Routing in WSN, *Journal of Internet Technology*, Vol. 19, No. 1, pp. 115-125, January, 2018.
[17] A. Saifullah, Y. Xu, C. Lu, Y. Chen, End-to-End Delay Analysis for Fixed Priority Scheduling in WirelessHART Networks, *IEEE Real-Time and Embedded Technology and Applications Symposium* Chicago, IL, 2011, pp. 13-22.
[18] X. Jin, J. Wang, P. Zeng, End-to-end Delay Analysis for Mixed-criticality WirelessHART Networks, *IEEE/CAA Journal of Automatica Sinica*, Vol. 2, No. 3, pp. 282-289, July, 2015.
[19] A. Saifullah, Y. Xu, C. Lu, Y. Chen, End-to-End Communication Delay Analysis in Industrial Wireless Networks, *IEEE Transactions on Computers*, Vol. 64, No. 5, pp. 1361-1374, May, 2015.
[20] G. Han, L. Liu, J. Jiang, L. Shu, G. Hancke, Analysis of Energy-Efficient Connected Target Coverage Algorithms for Industrial Wireless Sensor Networks, *IEEE Transactions on Industrial Informatics*, Vol. 13, No. 1, pp. 135-143, February, 2017.
[21] J. H. Lee, L. H. Kim, T. Kwon, FlexiCast: Energy-Efficient Software Integrity Checks to Build Secure Industrial Wireless Active Sensor Networks, *IEEE Transactions on Industrial Informatics*, Vol. 12, No. 1, pp. 6-14, February, 2016.
[22] X. Y. Yin, X. S. Zhou, Z. G. Li, S. N. Li, Joint Rate Control and Power Control for Lifetime Maximization in Wireless Sensor Networks, *Journal of Internet Technology*, Vol. 12, No. 1, pp. 69-78, January, 2011.
[23] A. Cerpa, J. L. Wong, M. Potkonjak, D. Estrin, Temporal Properties of Low Power Wireless Links: Modeling and Implications on Multi-hop Routing, *ACM International Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc'05)*, Urbana-Champaign, IL, 2005, pp. 414-425.
[24] K. Srinivasan, M. A. Kazandjieva, S. Agarwal, P. Levis, The β-factor: Measuring Wireless Link Burstiness, *International Conference on Embedded Networked Sensor Systems (SENSYS 2008)*, Raleigh, NC, 2008, pp. 29-42.
[25] M. H. Alizai, O. Landsiedel, K. Wehrle, Exploiting the Burstiness of Intermediate-Quality Wireless Links, *International Journal of Distributed Sensor Networks*, Vol. 8, No. 3, Article ID 826702, March, 2012.
[26] S. Montero, J. Gozalvez, M. Sepulcre, Link Scheduling Scheme with Shared Links and Virtual Tokens for Industrial Wireless Sensor Networks, *Mobile Networks & Applications*, Vol. 22, No. 6, pp. 1083-1099, December, 2017.
[27] S. Munir, S. Lin, E. Hoque, S. M. S. Nirjon, J. A. Stankovic, K. Whitehouse, Addressing Burstiness for Reliable Communication and Latency Bound Generation in Wireless Sensor Networks, *9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN 2010)*, Stockholm, Sweden, 2010, pp. 303-314.
[28] T. Huang, D. Li, Z. Zhang, L. Cui, Design of a Burst-link-aware Routing Protocol in Wireless Sensor Networks, *IET International Conference on Wireless Sensor Network, IET-WSN 2010*, Beijing, China, 2010, pp. 374-379.
[29] D. Yang, M. Gidlund, W. Shen, Y. Xu, T. Zhang, H. Zhang, CCA-Embedded TDMA Enabling Acyclic Traffic in Industrial Wireless Sensor Networks, *Ad Hoc Networks*, Vol. 11, No. 3, pp. 1105-1121, May, 2013.

Biographies

Hongchao Wang received the B.S. degree in communication engineering and the Ph.D. degree in communication and information systems from Beijing Jiaotong University, Beijing, China, in 2005 and 2012, respectively. Now, he works in the School of Electronic and Information Engineering, Beijing Jiaotong University. His research interests include Internet architecture, network security, and wireless sensor networks.

Jian Ma received his B.S degree in communication engineering from Beijing Jiao University, Beijing, China, in 2013. Currently, he is working toward the Ph.D degree in communications and information science from Beijing Jiaotong University, Beijing, China. His specific areas of research interest mainly focus on low-power network technologies and industrial wireless sensor networks.

Youzhi Xu received the B.S. degree from Xi'an Jiaotong University, Xi'an, China, and the Tech. Licentiate, Ph.D., and Docent degrees from Linkoping University, Linkoping, Sweden, in 1989, 1991, and 2000, respectively, all in electrical engineering. He is a Professor of data communication with the Department of Information Technology, Mid Sweden University, Sundsvall, Sweden. His research interests include video compression, source coding, channel coding, and communication networks.

Dong Yang received the B.S. degree from Central South University, Hunan, China, in 2003 and the Ph.D. degree in communications and information science from Beijing Jiaotong University, Beijing, China, in 2009. Currently, he works as an Professor in the School of Electronic and Information Engineering, Beijing Jiaotong University. His research interests include routing, Internet architecture, and wireless sensor networks.

Hongke Zhang received the M.S. and Ph.D. degrees in electrical and communication systems from the University of Electronic Science and Technology of China, Chengdu, China, in 1988 and 1992, respectively. Since 1994, he has been in BJTU. Now he is a Professor of the School of Electronic and Information Engineering. His specific areas of research interests include communications, computer networks, and information theory.