Optimal convergecast scheduling for hierarchical wireless industrial systems: performance bounds and two-stage algorithms

Xiaoling Zhang, Wei Liang, Haibin Yu, Xisheng Feng
Chinese Academy of Sciences, Shenyang Institute of Automation, Shenyang, Liaoning 110016, People’s Republic of China
E-mail: weiliang@sia.cn

Abstract: Increased mobility coupled with a possible reduction of cabling costs and deployment time makes wireless communication an attractive alternative for the industrial process monitoring and control. The major obstacles towards the utilisation of wireless industrial systems are predominantly the timing and reliability requirements. In this study, the authors take jointly the timing and reliability requirements, limited wireless resources and the cyclic data feature into consideration, and study the performance bounds and two-stage time- and channel-optimal convergecast scheduling algorithms for wireless industrial systems with hierarchical star and mesh architecture. Specifically, they consider the convergecast communication for wireless industrial systems operating according to the recent wireless network for industrial automation–process automation standard; and they will provide bounds on the minimum convergecast schedule length and bounds on the minimum number of channels for cluster-line and cluster-tree routing structures. In both cases, they propose time- and channel-optimal two-stage scheduling algorithms. They evaluate the author’s two-stage scheduling algorithms by both simulation and real hardwares. Numerical results demonstrate that their algorithms are efficient compared with traditional time division multiple access-based convergecast scheduling algorithms.

1 Introduction

With the success of wireless technologies in consumer electronics, standard wireless technologies are envisioned for the deployment in industrial environments to improve the functionality and the efficiency, which boost the formation of the industrial wireless systems (IWSs). The benefits of using IWSs are manifold [1]. The cost and time needed for the installation and maintenance of the large number of cables normally required in industrial environments can be substantially reduced. This is especially important in harsh environments where chemicals, vibrations, or moving equipments exist that could potentially damage any sort of cables. Stationary systems can be coupled wirelessly to any mobile subsystems or mobile robots in order to achieve a connectivity that would otherwise be impossible. Recently, some international organisations are actively promoting the standardised process of IWSs and have achieved several productions, such as WirelessHART [2] and wireless network for industrial automation–process automation (WIA-PA) [3]. However, numerous sources of radio-frequency and electromagnetic interference in industrial environments may cause communication challenges, leading to the risk of causing severe problems for applications with strict timing and reliability requirements [1, 4, 5]. Firstly, delayed or lost data may cause industrial applications to malfunction [6]. In addition, wireless communication may experience an excessive long latency because of the limited wireless frequency resource. Moreover, the collected data in most of IWSs are periodic and should be processed within its data update rate.

In view of the rigorous timing and reliability constraints, an appropriate medium access method, which addresses how to resolve potential contentions and collisions when using the wireless medium, is needed to adapt existing wireless technologies and protocols to industrial settings, or, when this is not sufficient, to develop new ones [7]. Existed medium access control (MAC) protocols utilising the back-off mechanisms and retransmission mechanisms would result in long latency and more collisions, while the time division multiple access (TDMA) method that could greatly eliminate collisions and obtain a bound on the time required to complete communication becomes the preferred choice. Scheduling is the base technique for TDMA. Typically, transmission scheduling is associated with wireless communication systems where time is slotted into intervals of fixed duration (namely timeslots) and frequency is equally divided into channels. Each wireless device is allocated timeslots and the corresponding channels for transmissions.

This paper considers the transmission scheduling for a critical functionality operating according to the recent WIA-PA standard in IWSs, so called convergecast. In most of IWS applications, such as energy scanning and temperature monitoring [8], many sensors are required to
deliver the collected data to one central control device. These many-to-one transmissions are generalised as convergecast. Our goal is to construct a framework, which contains both the theoretical analysis and algorithm of the time- and channel-optimal convergecast scheduling for IWSs with hierarchical star and mesh topology.

The rest of this paper is organised as follows. First, related work is introduced in Section 2, and in Section 3 we provide a generic framework, which encompasses the most relevant aspects of the WIA-PA networks and the problem formulation of convergecast scheduling. The time- and channel-optimal convergecast scheduling for networks with cluster-line routing and cluster-tree routing are studied in Sections 4 and 5. Section 6 presents simulation and experiment results and this paper is concluded in Section 7.

2 Related works

Real-time convergecast scheduling is a critical issue for IWSs. However, the time-constrained scheduling problem is NP-complete even in linear wireless systems [9–11]. Time-optimal convergecast scheduling, in which all devices in the network send data to a central control device in minimum time, is a special case of the general time-constrained scheduling problem [12, 13]. Moreover, the many-to-one communication brings more collisions during the convergecast delivery. When a great much of data surges to the central control device, devices near the central control device will become the ‘bottlenecks’ and cause serious congestion. There have been several research works dealing with the efficient TDMA convergecast scheduling for wireless sensor/ad-hoc networks. Choi et al [14] shown that convergecast scheduling problem is NP-hard in a weak sense. Kesselman and Kowalski [15] considered the problem of convergecast in ad-hoc geometric networks, where devices are located in a Euclidean plane. They assume that devices have a special collision detection capability so that a transmitting device can detect a collision within its transmission range. In addition, they also assume that one timeslot is long enough to allow multiple packet transmissions. Finally, they proposed a randomised distributed algorithm that has the expected running time $O(\log n)$. Gandham et al [16] proposed a near-optimal distributed convergecast scheduling algorithm for a tree network that requires at most $3N$ timeslots, where $N$ represents the number of devices in a network. Through extensive simulation, they demonstrate that the actual number of timeslots needed are about $1.5N$. Sun and co-workers [8] proposed a distributed convergecast scheduling algorithm, in which every device was scheduled by itself with information of one-hop neighbour devices and the total number of timeslots needed to complete the convergecast once was about $1.6N$ to $1.8N$. However, the mentioned-above works assumed that all devices communicate using a single channel. Recently, proposals for multi-channel convergecast scheduling methods have started to emerge [12, 17–20]. Most of them are designed to be used in the WirelessHART networks. Zhang et al. [12] studied the problems of optimal link scheduling and channel assignment for convergecast in the WirelessHART networks with line topology, and extended the results to tree topology in [20]. Soldati et al. [18] developed a novel mathematical programming framework for joint routing and link scheduling of deadline-constrained traffic in WirelessHART networks, and established lower bound on the evacuation time for line, multi-line and binary tree networks. Furthermore, Soldati completed the theoretical analysis and algorithm design for WirelessHART convergecast scheduling in [17]. To sum up, the centralised methods of convergecast scheduling require the global information of network and have high time complexity. While the distributed methods require much information, such as the number/length of branches in a network and relationship among nodes and branches, whose time complexity is equivalent to that of the centralised methods.

Existing analysis and algorithms for convergecast scheduling are not completely suitable for the WIA-PA networks because of the following characteristics. Firstly, compared to the mesh topology of WirelessHART networks and most of the existing research, the topology of WIA-PA network is hybrid star and mesh. Conclude the authors in [21], when high predictability of performance guarantees is the objective, it is suitable to rely on infrastructure-based wireless networks, such as hierarchical topology that is a combination of star and mesh, while the WIA-PA architecture follows this kind. Different topologies require a different allocation and dissemination methods for convergecast scheduling. Secondly, only field devices in the WIA-PA networks collect data to save energy, while all devices except for the gateway device collect data in [8, 12, 14–19]. The different data feature is another consideration during the convergecast scheduling process. In addition, we also take the data update rate into consideration. Our study follows the line in [12] for WirelessHART. For example, both kinds of networks utilise the scarce channel resource based on IEEE STD 802.15.4–2006 standard [22], demanding the convergecast schemes should be efficient in terms of channel utilisation. Moreover, the field devices in both kinds of networks are generally memory-constrained, and the convergecast schemes should be efficient in terms of memory-efficient.

In particular, this paper presents the following contributions:

- For a hierarchical star and mesh network with cluster-line routing, we analyse and prove the performance bounds for completing the convergecast transmission. We establish the lower bounds on the number of timeslots and channels based on three scenarios; all the Data Update Rates (DURs) in a cluster are same, all the DURs in a cluster are different, and the DURs in a cluster are partly same.
- For a hierarchical star and mesh network with cluster-tree routing, the same problem as the first item is studied.
- We propose the time- and channel-optimal two-stage scheduling algorithms for convergecast communications in hierarchical IWSs, which can be used for different routing topologies and different DUR scenarios.

3 Framework overview

The system model, assumptions and research problem are defined in this section.

3.1 Hierarchical star and mesh network model

The WIA-PA network extends the IEEE STD 802.15.4-2006 star topology to the hierarchical star and mesh topology, which is shown in Fig. 1. In the mesh topology, routing devices communicate by using multiple paths, which offers a high grade of reliability; in the star topology, one routing
device and several field devices/handheld devices constitute one cluster, in which the routing device acts as a cluster head and the field devices/handheld devices act as the cluster members. The routing devices forward data from field devices, which greatly reduces the energy consumption of field devices.

3.2 System model

We assume that all devices in the WIA-PA network are identified by the unique IDentiﬁers (IDs) and have same communication range. The gateway device has the network information about network connection and two-hop neighbouring devices of each device. This information can be collected by the following procedure: devices report the information on neighbouring devices to the gateway device with reporting periods proportional to IDs.

We model the WIA-PA network as an undirected graph $G=(V,E)$, where $V=\{GW,R,F\}$ represents all the network devices. GW denotes the gateway device; $R$ is the set of routing devices; and $F$ is the set of field devices. $E$ denotes the set of links, where two devices that can communicate successfully constitute one link. We assume that all links can transmit reliably. The unreliable communication is our future work. The DUR of a field device is $2^n$, where $n$ is a natural number. Specially, we assume that field devices have single-packet memory. One packet must be sent out before next new packet. In contrast, the routing devices need a multi-packet buffer to store packets from their clusters and other routing devices. Each device is equipped with a half-duplex transceiver, which implies that devices cannot transmit and receive simultaneously. Time is synchronised and slotted, and the length of one timeslot allows once a data transaction (transmitting exactly one packet and associated acknowledgement). A collection of timeslots repeating constitute one superframe. The WIA-PA superframe structure extends the IEEE STD 802.15.4-2006 superframe, and divides the inactive period into intra-cluster communication period, inter-cluster communication period and sleep period. The extended superframe structure is shown in Fig. 2.

3.3 Problem formulation

One field device initially generates at most one packet destined to the gateway device every DUR time during each convergecast. Packets are routed along $T=(V,E_r)$, where $E_r \subseteq E$ is the set of routing paths. For every device $i$, $R(i)$ and $O(i)$, respectively, denote the set of links that arrive at or leave device $i$; $p_d(i)$ denotes the capability of buffer after timeslot $t$, where we defined that $p_d(i) = 1$. Let $g_{ij}$ be the number of packets of a cluster with $R_{ij}$ being the cluster head in one superframe cycle. $g_{ij}$ is decided by the DURs of all field devices in a cluster. Let $L_s$ be the superframe length (in timeslots). Let $x_{ijt}$ be a binary variable indicating whether timeslot $t$ has been allocated to link $(i,j)$. If timeslot $t$ has been allocated to link $(i,j)$, $x_{ijt} = 1$; otherwise, $x_{ijt} = 0$.

The time- and channel-optimal scheduling to complete convergecast with the objective to minimise both the number of timeslots and the number of channels can be informally formulated as [17].

Problem 1: Time-optimal convergecast scheduling

\begin{equation}
\text{minimise } L_s
\end{equation}
Problem 2: Time- and channel-optimal convergecast scheduling

\[ \text{minimise } C_s = \max_{r \in [0, L_s]} \sum_{(i,j) \in E_r} x_{ij} \]  

s.t. 

\[ L_s = \text{calculated by (1)} - (5) \]  

where \( C_s \) is the number of channels used for transmission.

4 Time- and channel-optimal convergecast scheduling with cluster-line routing

Cluster-line routing, shown in Fig. 3, is instrumental for more general routings, and is the preferred topology in certain applications such as pipeline monitoring and unmanned offshore gas production [17]. Without loss of generality, the GateWay device (GW) is placed at the right end of the line, and \( N \) routing devices \( R_{11} \ldots R_{1N} \) are placed from right to left. For routing device \( R_{1i} \), \( i \in [1, N] \), it acts as a cluster head and manages several field devices. \( f_{1i} \) denotes the number of field devices in the cluster with \( R_{1i} \), being cluster head; and \( F_{1j} \) denotes the \( j \)th field device, \( j \in [0, f_{1i}] \); \( C_{1j} \) denotes the DUR of \( F_{1j} \). The following content studies three scenarios: all the DURs in a cluster are same, all the DURs in a cluster are different and the DURs in a cluster are partly same, which are briefly denoted as ADS, ADD and PDS.

Note that we have studied all the following theorems and corollaries in [23], and the detailed analysis and proof can refer to [23].

4.1 Scenario I: ADS

Considering the single-packet memory, the generated data must be sent out before the end of each superframe. Otherwise, the old data will be overlapped by newly generated data.

Owing to the half-duplex radio transceiver, the length of intra-cluster convergecast scheduling should not be less than the number of field devices. For network scalability, we assume that the intra-cluster communication periods of all clusters are equal, which is defined as \( \max_{i \in [1, N]} f_{1i} \). Each field device is assigned one timeslot during the intra-cluster communication period. In one cluster, because different field devices using different timeslots, using one channel can eliminate collisions completely.

Theorem 1: If all the data update rates of field devices in a cluster are same, that is \( C_{11} = C_{12} = \ldots = C_{1f_{1i}} \), and \( i = 1, 2, \ldots, N \), the lower bound on the number of timeslots used for the inter-cluster convergecast in a network with cluster-line routing is

\[ L_s = 2 \sum_{i=2}^{N} f_{1i} + f_{11} \]

Theorem 2: Given any scheduling algorithm \( S \) that can complete inter-cluster convergecast in

\[ L_s = 2 \sum_{i=2}^{N} f_{1i} + f_{11} \]

timeslots in a network with cluster-line routing, the lower bound on the number of channels used in \( S \) is

\[ \left[ \frac{2 \sum_{i=2}^{N} f_{1i} + f_{11} + 1}{2} - \left( \frac{2 \sum_{i=2}^{N} f_{1i} + f_{11} + 1}{2} \right)^{2} \right] \sum_{i=1}^{N} f_{1i} \]

From Theorems 1 and 2, we can discover that the results deduced in [12, 17] is special cases of our work when all the data update rates of field devices in a cluster are same.

4.2 Scenario II: ADD

Theorem 3: Given a cluster with more than one field device, if all the data update rates of field devices are different, that is, \( C_{1j} \neq C_{1k} \) \( \forall j, k \in [1, f_{1i}], j \neq k \) and \( i \in [1, N] \), the lower bound on the number of timeslots used for the intra-cluster convergecast in a network with cluster-line routing is 2.

Corollary 1: If all the data update rates of field devices in a cluster are different, that is, \( C_{1j} \neq C_{1k} \) \( \forall j, k \in [1, f_{1i}], j \neq k \) and \( i \in [1, N] \), the lower bound on the number of timeslots used for the inter-cluster convergecast in a network with cluster-line routing is \( 4N - 2 \), where \( N \) is the number of routing devices in a network.
Corollary 2: Given any scheduling algorithm \( S \) that can complete inter-cluster convergecast in \( 4N - 2 \) timeslots in a network with cluster-line routing, the lower bound on the number of channels used in \( S \) is

\[
\left[ \frac{4N - 1}{2} + \sqrt{\frac{12N^2 - 12N + 1}{4}} \right]
\]

where \( N \) is the number of routing devices in a network.

4.3 Scenario III: PDS

Theorem 4: Given a cluster with more than one field device and the data update rates of field devices in a cluster with \( R_i \) being cluster head are partly same, that is \( C_{i j} = C_{i k}, \forall j, k \in [1, f_i], j \neq k, i \in [1, N] \), the lower bound on the number of timeslots used for the intra-cluster convergecast in a network with cluster-line routing is \( \sum_{j=1}^{Q_{i j}} y_{i j} \). \( N_{i j} \) is the number of field devices with DUR being \( C_{i j} \) and satisfies \( \sum_{j=1}^{Q_{i j}} N_{i j} = f_i (Q_{i j} \leq f_i) \); \( Q_{i} \) is the number of DUR kinds in this cluster; and \( y_{i j} \) satisfies

\[
y_{i 1 j} = \left[ \frac{N_{i j} \times C_{i 1}}{C_{i j}} \right] = \left[ N_{i 1 j} \right] \tag{8}
y_{i 2 j} = \left[ \frac{N_{i 2 j} \times C_{i 1}}{C_{i 2}} \right] \tag{9}
y_{i k j} = \begin{cases} 
0, & \frac{C_{i 1} (N_{i (k-1)j} C_{i k} + N_{i k} C_{i (k-1)j})}{C_{i (k-1)j} C_{i k}} < y_{i (k-1) j} \\
\left[ \frac{N_{i k} - \left( \frac{y_{i (k-1) j} C_{i (k-1)j}}{C_{i 1}} - N_{i (k-1)j} \right) C_{i k}}{C_{i (k-1)j} C_{i 1}} \right], & \text{else}
\end{cases}
\tag{10}
\]

where \( k = 3, 4, \ldots, Q_{i} \).

Corollary 3: If the data update rates of field devices in a cluster are partly same, that is \( C_{i j} = C_{i k}, \forall j, k \in [1, f_i], j \neq k, i \in [1, N] \), the lower bound on the number of timeslots used for the inter-cluster convergecast in a network with cluster-line routing is

\[
L_s = 2 \sum_{i=2}^{N} \sum_{j=1}^{Q_{i j}} y_{i j} + \sum_{j=1}^{Q_{i 1 j}} y_{i 1 j}
\]

where \( N \) is the number of routing devices in a network and \( Q_{i} \) is the number of DUR kinds in a cluster with \( R_i \) being the cluster head.

Corollary 4: Given any scheduling algorithm \( S \) that can complete inter-cluster convergecast in

\[
\left[ \frac{2 \sum_{i=2}^{N} \sum_{j=1}^{Q_{i j}} y_{i j} + \sum_{j=1}^{Q_{i 1 j}} y_{i 1 j}}{2} + 1 \right] - \sqrt{\left( \frac{2 \sum_{i=2}^{N} \sum_{j=1}^{Q_{i j}} y_{i j} + \sum_{j=1}^{Q_{i 1 j}} y_{i 1 j}}{2} + 1 \right)^2 - \sum_{i=2}^{N} \sum_{j=1}^{Q_{i j}} y_{i j}}
\]
timeslots in a network with cluster-line routing, the lower bound on the number of channels used in \( S \) is (see equation at the bottom of the page)

where \( N \) is the number of routing devices in a network and \( Q_{i} \) is the number of DUR kinds in a cluster with \( R_i \) being the cluster head.

4.4 Design of two-stage convergecast scheduling algorithms

Taking into consideration of the hierarchical network topology, we propose a two-stage scheduling algorithm. The two-stage convergecast scheduling includes the inter-cluster convergecast scheduling and the intra-cluster convergecast scheduling. The basic idea is to allocate as many parallel transmissions as possible at each timeslot to capitalise on the available channels. For Scenario I ADS, we design a time- and channel-optimal scheduling algorithm called two-stage real-time convergecast scheduling with same data update rates (TRCS) based on the theoretical analysis of the performance limits. The inter-cluster convergecast scheduling (first-stage) is executed by the \( GW \). The scheduling results for inter-cluster communications are disseminated to all routing devices; then, the intra-cluster convergecast scheduling (second-stage) is executed by the routing devices after receiving the corresponding sub-schedules. The TRCS algorithm is realised by running following Algorithms 1-4. The detailed implement of Algorithms 1-4 is introduced as follows:

- **First stage: inter-cluster scheduling:** We improve the algorithm ‘ConvergeCast_Line Multi-packet buffering’ in [17] to be adaptive to our situations. Compared to one packet in [17], the number of transmitted packets by each routing device in Scenario I is equal to the number of field devices in the local cluster. We design five scheduling rules to schedule multiple packets in one routing device and compute the inter-cluster convergecast scheduling from the farthest routing devices through two steps: forward scheduling and backward scheduling.

Rule 1: Using timeslots and channels as few as possible, which are bounded in Theorems 1 and 2;

Rule 2: Less than \( PT_{\text{max}}(t) \) routing devices are scheduled to transmit in timeslot \( t \), in which \( PT_{\text{max}}(t) \) [23] denotes the maximum number of parallel transmissions scheduled in timeslot \( t \);

Rule 3: Starting schedule from routing device farthest from \( GW \), and terminating till all packets have been transmitted to \( GW \);

Rule 4: Forward scheduling and backward scheduling;

(a) **Forward scheduling:** if \( R_i, (i \in [1, N]) \) is assigned timeslot \( t \) and channel \( c \) for transmission and \( R_{i (t-2)} \) \( (i \in [3, N]) \) has packets in its buffer, \( R_{i (t-2)} \) is scheduled for transmission
by using timeslot $t$ and channel $c + 1$; Otherwise, routing devices from $R_{(i-3)}$ to $R_1$ are judged as following the same way as $R_{(i-2)}$ subsequently.

(b) Backward scheduling: if the routing devices $R_{(i-2)} - R_{11}$ have no packets in their buffers or the number of devices scheduled is less than the maximum transmissions that can be scheduled in this timeslot during the forward scheduling step, routing devices from $R_{(i+2)}$ to $R_{1N}$ are judged subsequently.

Rule 5: Allocating timeslots and channels for whole path from each routing device to $GW$ in an orderly sequence.

The detailed inter-cluster convergecast scheduling algorithm is given in Algorithm 1. $N_t(R_i)$ denotes the number of packets of routing device $R_i$ in timeslot $t$. The function of SearchFirstRouter in Algorithm 1 is given in Algorithm 2, which aims to find the starting device for inter-cluster scheduling. Joint forward scheduling and backward scheduling can maximise the number of transmitting devices in each timeslot. Since the time complexity for both steps are $O(N)$, the time complexity of Algorithm 1 is $O(N^2)$.

After completing the inter-cluster convergecast scheduling, $GW$ disseminates the scheduling results to all routing devices. The inter-cluster convergecast scheduling is stored in a compact structure $S[t][ch]$ that records the timeslots and channels used by routing devices for transmissions. $S[t]$ records the scheduling results in timeslot $t$ on all channels. Once a routing device receives the $S[t][ch]$, it should extract its sub-schedule. The sub-extraction method (Algorithm 3) can refer to the ‘sub-scheduling and channel hopping sequence generation’ algorithm in [17]. In each timeslot, a device can work in three states: transmitting (T), receiving (R) and sleep (S). Each routing device stores the sub-schedule in a two-dimensional (2D) array $S_s[t][2]$, where $S_s[t][1]$ records the state of a routing device in timeslot $t$ and $S_s[t][2]$ records the channel offset. The time complexity of Algorithm 3 is $O(N_C^2)$.

Fig. 4 gives an example of the inter-cluster convergecast scheduling in a network with four clusters. $A \rightarrow B$ denotes that device $A$ sends a packet to device $B$. According to Theorems 1 and 2, the lower bounds on the number of timeslots and channels for inter-cluster convergecast are, respectively, 18 and 2. After receiving the inter-cluster scheduling $S[t][2]$, $R_{13}$ extracts its sub-schedule $S_s[t][2]$ by using Algorithm 3.

- Second stage: intra-cluster scheduling: The intra-cluster scheduling is executed by a routing device for allocating timeslots and channels to field devices in one cluster. The basic idea of the intra-cluster scheduling is to search idle timeslots starting from the field device with the smallest ID. This paper assumes that all devices in a cluster use same channels and neighbouring clusters use different channels for avoiding collisions. The channel assignment method of neighbouring clusters can be realised by using the mostly familiar colouring algorithm as in [24].

The intra-cluster convergecast scheduling algorithm is given in Algorithm 4. In Algorithm 4, $L_{sa}$ denotes the number of timeslots of intra-cluster communication period. $t [L_{sa}]$ indicates the timeslot status with 1 denoting the timeslot being occupied and 0 otherwise. The time complexity of Algorithm 4 is $O((\max_{i \in [1, N]} t_{fi})^2)$.
For Scenario II, the two-stage real-time convergecast scheduling with different data update rates (TRCD) algorithm modifies the TRCS algorithm to accommodate the different DURs and allows multiplex of timeslots. As the basic idea of the first-stage scheduling is similar with that of Scenario I except for the number of timeslots and channels, we will focus on the method of intra-cluster convergecast scheduling. The basic idea of intra-cluster convergecast scheduling for Scenario II is to utilise at most two timeslots for convergecast by all field devices in a cluster. The detailed intra-cluster convergecast scheduling algorithm is given in Algorithm 5, the implement of which is similar as that of Algorithm 1. In Algorithm 5, $\mathbb{I}[L_m]$ is a binary array that is used to indicate the status of a timeslot. If the timeslot is idle, the corresponding element is 0; otherwise, the corresponding element is 1. $\mathcal{S}$ denotes the basic unit of superframe length (in timeslots), which is defined by users.

For Scenario III, the two-stage real-time convergecast scheduling with partly same data update rates (TRCP) algorithm modifies the TRCS algorithm to accommodate the partly same DURs and allows partly multiplex of the timeslots. As in Scenario II, we focus on the intra-cluster convergecast scheduling. The basic idea is to utilise at most $\sum_{i=1}^{2} y_{13i}$ timeslots calculated in Theorem 4 to finish the intra-cluster convergecast. The detailed intra-cluster convergecast scheduling algorithm is given in Algorithm 6, the implement of which is similar as that of Algorithm 1.

In Algorithm 6, the variable idle_slot_number is used to indicate the starting number of timeslots.

Fig. 5 gives an example of Algorithm 6. The numbers written above devices are correspondent DURs. According to Theorem 4, the parameters for intra-cluster convergecast scheduling in the cluster with $R_{13}$ being the cluster head, respectively, are: $C_{131} = 1$, $C_{132} = 4$, $N_{131} = 2$, $N_{132} = 1$, $Q_{13} = 2$, $y_{131} = 2$ and $y_{132} = 1$. The total number of packets at the beginning of each superframe cycle is $S_{13} = \sum_{i=1}^{2} y_{13i} = 3$. The scheduling results are on the underside of Fig. 5.

### 5 Time- and channel-optimal convergecast with cluster-tree routing

Fig. 6 is a network with cluster-tree routing. There exist $N$ routing devices in the network, which manage several field devices and constitute clusters. Let $D$ denotes the depth of the routing tree. The $GW$ is located at level 0 and has $m$ subtrees $T_1, T_2, \ldots, T_m$ rooted at routing devices $R_{11}, R_{21}, \ldots, R_{m1}$, respectively. The number of routing devices in subtrees $T_1, T_2, \ldots, T_m$, respectively, are $n_1, n_2, \ldots, n_m$. Without loss of generality, it is assumed that $n_1 \geq n_2 \geq \cdots \geq n_m$. Therefore $T_1$ is the largest subtree. Let $R_i$ denotes the $i$th routing device in the $j$th subtree and $f_j$ be the number of field devices in the cluster with $R_i$ being the cluster head ($i \in [1, m]$, $j \in [1, n_i]$). Next, we will
investigate the convergecast scheduling problems in networks with cluster-tree routing from three scenarios: ADS, ADD and PDS.

5.1 Scenario I: ADS

Theorem 5: If all the data update rates of field devices in a cluster are same, the lower bound on the number of timeslots required to complete the inter-cluster convergecast in a network having cluster-tree routing and $N$ routing devices is

$$L_s = \max \left\{ 2 \sum_{j=2}^{n_l} f_{11} + f_{11}, \sum_{i=1}^{m} \sum_{j=1}^{n_l} f_{j} \right\}$$

where $f_{j}$ is the number of field devices in the cluster with $R_j$ being cluster head ($i \in [1, m], j \in [1, n_l]$).

Theorem 5 shows that the structure of the routing tree plays a fundamental role in minimising the convergecast time and dominates the minimum network latency. If

$$2 \sum_{j=2}^{n_l} f_{11} + f_{11} \leq \sum_{i=1}^{m} \sum_{j=1}^{n_l} f_{j} \quad \text{that is} \quad \sum_{j=2}^{n_l} f_{11} \leq \sum_{i=2}^{m} \sum_{j=1}^{n_l} f_{j}$$

then the lower bound on the number of timeslots to complete the inter-cluster convergecast is

$$L_s = \sum_{i=1}^{m} \sum_{j=1}^{n_l} f_{j}$$

which is equal to the number of packets in the whole network. Otherwise, the lower bound on the number of timeslots to complete the inter-cluster convergecast is

$$L_s = 2 \sum_{j=2}^{n_l} f_{11} + f_{11}$$

which is equal to the number of packets in the largest subtree $T_1$. Finding a minimum spanning tree subject to cardinality constraints on the number of routing devices in any subtree, the so called capacitated minimum spanning tree problem, is known to be NP-hard [17], and the effective heuristics algorithm should be carefully designed.

Theorem 6: Given any scheduling algorithm $S$ that can complete inter-cluster convergecast in

$$L_s = \max \left\{ 2 \sum_{j=2}^{n_l} f_{11} + f_{11}, \sum_{i=1}^{m} \sum_{j=1}^{n_l} f_{j} \right\}$$

timeslots in a network with cluster-tree routing, the lower bound on the number of channels used in $S$ is

$$\left[ (L_s + 1/2) - \frac{(L_s + 1/2)^2 - 2 \sum_{d=1}^{D} (d \times n(d))}{2} \right]$$

where $n(d)$ denotes the total number of packets in routing devices at level $d$.

5.2 Scenario II: ADD

In Scenario II, the DURs of field devices in the cluster with $R_j$ ($i \in [1, m], j \in [1, n_l]$) being cluster head satisfy the following relationship

$$C_{ij} \neq C_{ij}, \ \forall k, l \in [1, f_{ij}], \ k \neq l$$

(11)

Theorem 7: Given a cluster with more than one field device, if all the data update rates of field devices in a cluster are different, that is, $C_{ij} \neq C_{ij} \forall k, l \in [1, f_{ij}], \ k \neq l$ and $i \in [1, m], j \in [1, n_l]$, the lower bound on the number of timeslots used for the intra-cluster convergecast in a network with cluster-tree routing is 2.

For ADD, the number of packets in a routing device is equal to the number of timeslots for intra-cluster convergecast. According to Theorem 7, at the beginning of the inter-cluster convergecast, the number of packets in a routing device is 2.

Corollary 5: If all the data update rates of field devices in a cluster are different, that is, $C_{ij} \neq C_{ij} \forall k, l \in [1, f_{ij}], \ k \neq l$ and $i \in [1, m], j \in [1, n_l]$, the lower bound on the number of timeslots used for the inter-cluster convergecast in a network with cluster-tree routing is

$$L_s = \max \left\{ 4n_1 - 2, \sum_{i=1}^{m} n_i \right\}$$

where $n_i$ is the number of routing devices in subtree $T_i (i \in [1, m])$.

Corollary 6: Given any scheduling algorithm $S$ in a network with cluster-tree routing that can complete inter-cluster convergecast in

$$L_s = \max \left\{ 4n_1 - 2, \sum_{i=1}^{m} n_i \right\}$$

timeslots, the lower bound on the number of channels used in $S$ is

$$\left[ (L_s + 1/2) - \frac{(L_s + 1/2)^2 - 2 \sum_{d=1}^{D} (d \times n(d))}{2} \right]$$

where $n(d)$ denotes the total number of packets in routing devices at level $d$.

5.3 Scenario III: PDS

In Scenario III, the DURs of field devices in the cluster with $R_j$ ($i \in [1, m], j \in [1, n_l]$) being cluster head satisfy the following relationship

$$C_{ij} = C_{ij}, \ \exists k, l \in [1, f_{ij}], \ k \neq l$$

(12)

Corollary 7: Given a cluster with $R_j$ ($i \in [1, m], j \in [1, n_l]$) being cluster head, the data update rates of field devices in a cluster are partly same, that is $C_{ij} = C_{ij}, \ \exists k, l \in [1, f_{ij}], \ k \neq l$. Supposing that the number of field devices with DUR
being \( C_{ijk} \) is \( N_{ijk} \) and \( \sum_{j=1}^{Q_0} N_{ijk} = f_{ij} \) \( (Q_{ij} \leq f_{ij}) \), the lower bound on the number of timeslots used for the intra-cluster convergecast in a network with cluster-tree routing is 
\[
\sum_{j=1}^{Q_0} y_{ijk}, \quad \text{where } i \in [1, m], j \in [1, n_j], k \in [1, Q_{ij}], Q_{ij} \text{ is the number of DUR kinds in the cluster with } R_i \text{ being cluster head, and } y_{ijk} \text{ is defined as }
\]
\[
y_{ijk} = \left[ \frac{N_{ijk} \times C_{jik}}{C_{g1}} \right] = \left[ \frac{N_{ijk}}{C_{g1}} \right] (13)
\]
\[
y_{ij2} = \left[ \frac{N_{ijk} \times C_{jik}}{C_{g2}} \right] (14)
\]
(see (15))

where \( k = 3, 4, \ldots, Q_{ij} \).

Corollary 8: If the data update rates of field devices in a cluster are partly same, that is, \( C_{ijk} = C_{jik}, \forall k \in [1, Q_{ij}], k \neq l \), the lower bound on the number of timeslots used for the inter-cluster convergecast in a network with cluster-tree routing is
\[
L_s = \max \left\{ 2 \sum_{j=2}^{Q_0} \sum_{k=1}^{Q_{ij}} y_{ijk} + \sum_{j=1}^{Q_0} y_{jk1}, m \sum_{j=1}^{Q_0} \sum_{k=1}^{Q_{ij}} y_{ijk} \right\}
\]
where \( n_i \) is the number of routing devices in subtree \( T_i (i \in [1, m], j \in [1, n_j], k \in [1, Q_{ij}]) \).

Corollary 9: Given any scheduling algorithm \( S \) that can complete inter-cluster convergecast in
\[
L_s = \max \left\{ 2 \sum_{j=2}^{Q_0} \sum_{k=1}^{Q_{ij}} y_{ijk} + \sum_{j=1}^{Q_0} y_{jk1}, m \sum_{j=1}^{Q_0} \sum_{k=1}^{Q_{ij}} y_{ijk} \right\}
\]
timeslots in a network with cluster-tree routing, the lower bound on the number of channels used in \( S \) is
\[
\left( L_s + 1/2 \right) - \sqrt{\left( L_s + 1/2 \right)^2 - 2 \sum_{d=1}^{D} \left( d \times n(d) \right)}
\]
where \( n(d) \) denotes the total number of packets in routing devices at level \( d \).

### 5.4 Design of two-stage convergecast scheduling algorithms

Based on the performance limits on the number of timeslots and channels for completing both intra- and inter-cluster convergecast, we propose a two-stage real-time convergecast scheduling with same data update rates for cluster-tree routing topology (TRCS-T). The routing tree is variable with differences of depth and subtree scales. We aim to find a general algorithm that is effective to the changeable routing tree. The priority-based approach is suitable and effective for routings with tree topology [25].

The priority of each routing device is initialised combining the following six rules:

- Giving higher priority to subtree with more packets.
- Randomly sorting subtrees with same quantity of packets and allocating priority from big to small according to the sequence.
- Giving higher priority to routing device with deeper depth in the same subtree.
- Giving higher priority to routing device with more packets in the same subtree and at the same level.
- Randomly sorting routing devices with same quantity of packets at the same level in the same subtree and allocating priority from big to small according to the sequence.
- The largest value of the priorities is \( N \), which is equal to the number of routing devices in the network and bigger value means larger priority. The priority with 0 means that the scheduling has been completed.

The initialisation of priorities is illustrated in Fig. 6 and the numbers above the routing devices indicate the priorities.

The TRCS-T algorithm is implemented by two steps. The first step is to initialise priorities for all routing devices and the second step is to allocate timeslots and channels for each link. The detailed algorithm is given in Algorithm 7.

The time complexity of the first stage is \( O(N^2) \) and the time complexity of the second stage is \( O(\max_{i \in [1, m]} \sum_{j \in [1, n]} |f_{ij}|) \).

By executing TRCS-T algorithm, the network shown in Fig. 6 has the following two results of inter-cluster convergecast scheduling possibly, which is shown in Figs. 7a and 8. According to Theorems 5 and 6, the lower bounds on the number of timeslots and channels for inter-cluster convergecast scheduling in the network shown in Fig. 6 are 17 and 2, respectively. Fig. 7 indicates that the lower bound on the number of timeslots for inter-cluster convergecast may not be always achievable and the tightness of the lower bound depends on the routing structure and special algorithms. Certainly, the tightness of the time bound can be fulfilled by increase the number of channels, which also breaks the tightness of the lower bound on the number of channels.

The two-stage time- and channel-optimal convergecast scheduling algorithm in Scenarios II and III are similar to TRCS-T algorithm except for the search ranges during the first stage. In Scenario II, the search ranges of the first stage are bounded by Corollaries 5 and 6; in Scenario III, the search ranges of the first stage are bounded by Corollaries 8 and 9.

### 6 Performance evaluation

The optimality and overhead of the two-stage convergecast algorithms for networks with both cluster-line routing and
cluster-tree routing are evaluated through both simulations and experiments on real hardware. In addition, the results are extended to analyse the timeliness and network scale, which provides a guidance for network planning and actual applications.

6.1 Simulation analysis

We evaluate our algorithm through simulations in OPNET 10.0 network simulator. The network area is bounded in $100 \times 100$ m$^2$. The physical parameters are equal to those in IEEE STD 802.15.4-2006 standard. The maximum packet length is 128 octets. The transmitting power and the receiving power of the half-duplex transceiver are 15.16 and 35.28 mW, respectively. The bit rate is 250 kbps and the transmission radius is 25 m. The centre frequency of the 16 channels in 2.4 G band is defined as: $F_c = 2405 + 5(k-11)$ in megahertz for $k = 11, 12, \ldots, 26$, where $k$ is the channel number. The duration of a timeslot is set to 31.25 ms in this simulation. The DUR of each field device is configured randomly.

6.1.1 Performance of the two-stage algorithms with cluster-line routing: The effectiveness of the two-stage convergecast scheduling for three scenarios in the network with cluster-line routing are illustrated in Fig. 8. The simulation results are compared with the optimal values computed in Section 4. Meanwhile, the timeliness of two-stage scheduling algorithm for three scenarios are compared with the most typical convergecast scheduling algorithm in [17]. Simulation results show that the two-stage scheduling algorithms for networks with cluster-line routing are optimal, which indicates that the convergecast can be completed using the lower bound on the number of timeslots. However, the completion time of convergecast for three scenarios are different. The completion time for Scenario I is the longest compared with other two scenarios because of the un-reusability of timeslots, while the completion time of convergecast for Scenario II is the shortest because two timeslots are enough for intra-cluster convergecast regardless of the number of field devices in a cluster. This result provides a guidance for network configuration. To reduce the overall convergecast time, field devices with different DURs should join in one cluster and field devices with same DURs should join in different clusters. The maximal convergecast time for all scenarios are $0.37 \text{s}$ when the number of devices are 10 and $0.66 \text{s}$ when the number of devices are 16 compared with 0.594 and 0.906 s, respectively, in WirelessHART convergecast operations [17]. The above numerical results illustrate that the timeliness of the two-stage scheduling algorithm is better than the centralised scheduling algorithm. In addition, the overheads of time and control packets in two-stage scheduling algorithms are reduced by more than 6–8 and 2 times, respectively, compared with that of a centralised scheduling algorithm, which also validates that the two-stage scheduling is more effective and faster than the centralised scheduling.

6.1.2 Performance of the two-stage algorithms with cluster-tree routing: The scheduling length and the deviation for various topology parameters are simulated compared with the centralised heuristic algorithm for WirelessHART [17]. We use three parameters to represent the network with cluster-tree routing, which are denoted as $M$, $D$ and $O$. $M$ denotes the number of childrens of $GW$; $D$ denotes the depth of the tree; and $O$ denotes the number of field devices in the cluster. Each simulation setting is represented as $(M, D, O)$. Different combination of these three parameters will generate different topology.

We introduce the average deviation of simulation scheduling length from the lower bound [17] as

$$\text{AvgDiff} = (\hat{L}_a - L_a)/L_a$$

where $\hat{L}_a$ denotes the actual scheduling length and $L_a$ denotes the lower bound calculated in Section 5. The comparisons of average deviation and deviation frequency are shown in Fig. 9.

In Figs. 9a and b, AvgDiff achieved by TRCS-T algorithm is smaller than that of WirelessHART in [17]. AvgDiff of both algorithms change smoothly. The maximum AvgDiff achieved by a TRCS-T algorithm is below 0.45% compared with 2.5% achieved by the centralised scheduling algorithm for WirelessHART, which demonstrate the better near-optimality achieved by TRCS-T algorithm. Fig. 9c plots the maximum deviation of AvgDiff from the lower bound for $O = 2$ under different values of $M$ and $D$. Fig. 9d plots the occurrence frequency of the discrepancies. The simulation is run 100 times. To the centralised scheduling algorithm for WirelessHART, the maximum difference goes up to 15 timeslots that corresponds to less than 2.2% of the lower bound when $D = 10$ and the occurrence frequency is less than 0.04%. TRCS-T algorithm demonstrates its smaller deviation and lower occurrence frequency compared with the centralised scheduling algorithm. When $D = 10$, the maximum difference is eight timeslots that corresponds to less than 1.4% of the lower bound and the occurrence frequency is less than 0.016%.
From simulations, we can conclude that the two-stage algorithm performs better than the centralised scheduling algorithm in aspects of reducing protocol overheads and approaching optimality in networks utilising cluster-tree routing.
routing. The reason is that the two-stage algorithm is based on hierarchical star and mesh topology, where the scheduling is computed by more than one device, such as GW and routing devices. Therefore the computation complexity and scheduling dissemination time are reduced.

6.2 Experiment validation

We validate the two-stage convergecast scheduling algorithms from a real implementation on the SIA2420 modules [26]. SIA 2420 module is completely compliant to the IEEE STD 802.15.4-2006 physical layer and built in multiple specifications, such as WIA-PA, WirelessHART and Zigbee. The user interface is built on the Microsoft Visual C++ 6.0. The experiment platform, network topology and user operation interface are shown in Fig. 10. The DUR of each field device is configured randomly.

Fig. 11a compares the convergecast time with the optimal results for three scenarios in the network with the cluster-line routing topology. The experiment results show that there exist an optimal strategy for convergecast communication in the network with cluster-line routing topology. This conclusion is same as that of the simulation results. Fig. 11b compares the convergecast time with the optimal results for three scenarios in the network with the cluster-tree routing topology, which show that the two-stage scheduling algorithms are near-optimal and the maximum near-optimal ratio is 99.9%.

6.3 Extension and application

The theoretical analysis of the lower bounds on the number of timeslots and channels gives deep insights into the network planning and reliable scheduling algorithm designing. In order to increase the reliability for IWSs, a retransmission scheme is introduced.

We consider a network with the more general cluster-tree routing topology. According to the theoretical analysis in Section 5, the lower bound on the number of timeslots for intra-cluster convergecast in Scenario I is \( \max_{i \in [1,m], j \in [1,n_i]} f_{ij} \) and the lower bound on the number of timeslots for inter-cluster convergecast is

\[
\max \left\{ \sum_{j=2}^{m} f_{i1} + f_{i11}, \sum_{j=1}^{n_i} f_{ij} \right\}
\]

If we consider three retransmissions for each frame (default value of \textit{mac Max Frame Retries} in IEEE STD 802.15.4-2006), the lower bounds on the number of timeslots for intra- and inter-cluster convergecast in Scenario I are four times of

\[
\max_{i \in [1,m], j \in [1,n_i]} f_{ij} \quad \text{and} \quad \max \left\{ \sum_{j=2}^{m} f_{i1} + f_{i11}, \sum_{j=1}^{n_i} f_{ij} \right\}
\]

respectively. Meanwhile, the number of timeslots during the beacon, CAP and CFP periods are 16 according to the IEEE STD 802.15.4-2006. In all, the superframe length can

![Fig. 10 Experiment settings](image1)

\( a \) Experiment platform

\( b \) Network topology and monitoring data

![Fig. 11 Convergecast time by experiment](image2)

\( a \) Cluster-line routing

\( b \) Cluster-tree routing
be calculated as
\[
16 + 4 \times \max_{i \in [1, m]} \max_{j \in [1, n]} f_{ij} + 4 \\
\times \max \left \{ 2 \sum_{j=2}^{n_i} f_{i1} + f_{i1}, \sum_{i=1}^{m} \sum_{j=1}^{n_i} f_{ij} \right \} \times \text{duration of timeslot}
\]
For a network having 1000 field devices, that is \( \sum_{i=1}^{m} \sum_{j=1}^{n_i} f_{ij} = 1000 \), and the duration of the timeslot being 10 ms, if
\[
2 \sum_{j=2}^{n_i} f_{i1} + f_{i1}, \sum_{i=1}^{m} \sum_{j=1}^{n_i} f_{ij}
\]
the maximum superframe length is 80.16 s, which is allowed in most applications of the process industries. In Scenarios II and III, the superframe lengths are less than 80.16 ms because of smaller number of timeslots.

The analysis of superframe length indicates that the network with hierarchical star and mesh topology can accommodate more devices and can guarantee better timeliness. For applications required large network and tight timeliness, if we assign field devices with different DURs in the same cluster and utilise the two-stage algorithms, the requirements can be well satisfied.

7 Conclusion and future work
This paper studies several issues of time- and channel-optimal convergecast scheduling for industrial wireless systems with hierarchical topology which includes analysis of performance bounds and design of two-stage scheduling algorithms. We discuss these problems for two routing topologies: cluster-line and cluster-tree in three scenarios: ADS, ADD and PDS. For each setting, we establish the lower bounds on the number of timeslots and channels required for completing the intra- and inter-cluster convergecast and design effective two-stage algorithms for generating time- and channel-optimal schedules. Simulation and experiment results demonstrate that our algorithms are optimal or near-optimal, respectively, for networks with cluster-line and cluster-tree routing topologies. Finally, we extend our work to the applications that require reliable and real-time communication. An interesting issue to be investigated in the future work is the dissemination communications from the gateway device to other devices in the network.

8 Acknowledgments
This work was supported by the Key Laboratory of Networked Control Systems of Chinese Academy of Sciences, and supported by the Natural Science Foundation of China (61202418 and 61174026), the National High Technology Research and Development Program of China (863 Program: 2011AA040101), and Strategic Priority Research Program of the Chinese Academy of Sciences (XDA06021000).

9 References
26 SIA2420: http://www.industrialwireless.cn