IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 19, 2016, accepted January 11, 2017, date of publication January 19, 2017, date of current version June 7, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2654483

Bounding the Demand of Mixed-Criticality Industrial Wireless Sensor Networks

CHANGQING XIA[1], XI JIN[1], LINGHE KONG[1,2], AND PENG ZENG[1,3]

[1]State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, China
[2]Shanghai Jiao Tong University, Shanghai, China
[3]Shenyang Institute of Automation, Chinese Academy of Sciences, Guangzhou, China

Corresponding author: P. Zeng (zp@sia.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61502474, Grant 61233007, and Grant 61672349 and in part by the Youth Innovation Promotion Association CAS.

ABSTRACT Wireless sensor networks (WSNs) have been widely used in industrial systems. Industrial systems demand a high degree of reliability and real-time requirements in communications. In many industrial WSNs applications, flows with different levels of criticality coexist in the system. When errors or exceptions occur, high-criticality flows must be guaranteed reliably and in real time. However, only a few works focus on mixed-criticality industrial systems. Concerning this issue, in this paper, we study mixed-criticality industrial systems and propose a supply/demand bound function analysis approach based on earliest deadline first scheduling. In addition, our method considers both source routing and graph routing. At the beginning, when the system is in low-criticality mode, source routing considers the schedulability of each flow. When errors or exceptions occur, the system switches to high-criticality mode, and network routing turns to graph routing to guarantee that important flows can be scheduled. By estimating the demand bound for mixed-criticality systems, we can determine the schedulability of industrial systems. Experiments indicate the effectiveness and efficacy of our approach.

INDEX TERMS Industrial networks, scheduling, WSN, mixed-criticality, graph routing.

I. INTRODUCTION

Industrial wireless sensor networks are emerging as a new generation of communication infrastructure for industrial process monitoring and control [16], [28]. Compared to conventional process control systems, industrial wireless sensor networks have the potential to save costs and enhance reliability and flexibility. Based on the features of industrial wireless sensor networks, industrial standards such as WirelessHART [7], [11], ISA100 [1] and WIA [17] are used extensively. Fig 1 is a model of an auto production line, which is based on the WIA-FA standard.

Industrial networks adopt Wireless Sensor-Actuator Networks (WSANs) in which sensors and actuators communicate through low-power multi-hop wireless mesh networks [15], [28]. The delay caused by communication may lead to system performance degradation or even system error. Therefore, it is meaningful to effectively estimate the worst-case communication delay for real-time flows in industrial wireless sensor networks, especially flows that are relatively important. The existing literature focuses on single-criticality systems, and the importance of flows is reflected only in the priority. However, the priority cannot represent importance in many situations. For instance, there are image capture and motor controllers in unmanned aerial vehicle systems. For tasks such as image capture, which have high priority but low criticality, they require only the general quality certification. Tasks such as the motor controller require strict safety certification such as that of the Federal Aviation Administration (FAA) or European Aviation Safety Agency (EASA), although their priority is lower than image capture. Hence,

FIGURE 1. Auto production line.

we not only priority but also a few criticality modes are needed to ensure the reliability of the network. It is necessary to introduce mix-criticality concept into wireless networks. We take thr precision agriculture in WSNs as an example [14]. Several kinds of sensors are deployed in the fields. For regular tasks such as monitor and control the nutrition requirements of plants, they have high priority but low criticality. In contrast, tasks such as monitor plants diseases and insect pests, they have a higher level criticality although their priority is lower than monitor and control.

Reliability is one of the most important characters in industrial wireless sensor networks. Graph routing [7] as an effective way to improve network reliability has been widely used in recent years. A network under graph routing allocates two dedicated time slots for each transmission; if the first transmission fails, a retransmission will be sent. Furthermore, the controller assigns a third shared slot on a separate path for another retransmission. However, graph routing introduces great challenges for real-time analysis. Many conflicts are generated on a large number of transmission tasks. Obviously, the task which is more critical but has a low priority may miss its deadline in this situation. However, many systems need to guarantee high-level task's schedulability even though in the worst case. That is really very important in many circumstances such as industrial production line, vehicle driving system, etc. To improve high-critical flows schedulability and analyze the schedulability of system, we introduce mixed-criticality system and resource analysis into wireless industrial networks. Mixed-criticality system can improve high-level flow's schedulability by switching the system criticality, and resource analysis is a major way to analyze the schedulability in real-time systems. Combining mixed-criticality system and resource analysis, we can improve high-critical flows schedulability, and estimate system schedulability with different critical levels.

To improve the system reliability and high-criticality flow schedulability, in this study, we propose a novel industrial network model, which considers both mixed-criticality and network routing. Our objective is to network reliability, especially for high-criticality flows to arrive at their destinations on time even though in the worst case. We analyze network schedulability by the method of resource analysis. The network is reliable when network resource supply is no less than network upper demand in any length of time slot. The main challenges in our work are (1) how to evaluate network demand at the time slot system switches from low-criticality mode to high-criticality mode and (2) how to tight the network demand bound function to ensure that the result is not too pessimistic. At the beginning, the system works in low-criticality mode, and the flows transmit under source routing. The packets transmit from the source to the destination on the primary paths; when an error occurs or the demand changes, the system switches to high-criticality mode to enhance the schedulability of high-criticality flows. The network substitutes reliable graph routing for source routing. Furthermore, we present a supply/demand bound analysis method to analyze the schedulability of periodic flows in industrial wireless sensor networks. By comparing the relationship of network supply bound and demand, we can predict whether the network can be scheduled. The priority is assigned by the Earliest Deadline First (EDF) policy [18]. The current study makes the following key contributions:

1) We propose a mixed-criticality industrial system, in which network routing switches from source routing to graph routing when the criticality mode changes. To our best knowledge, this is the first analysis for mixed criticality under both source and graph routings.
2) We theoretically derive the supply/demand bound function as a novel analysis method for industrial wireless sensor networks. By analyzing channel contention and transmission conflict, we obtain the upper-bound function of demand in any length of time slot. When given a network supply bound function, we can determine the schedulability of flows under different criticality modes.
3) We tighten the demand bound by analyzing *carry-over jobs* (which are released but not finished at the switching slot) and discussing the number of conflicts between two flows.
4) Our method can be applied for general networks. By calculating the maximum demand bound of system, we can analyze network schedulability in the system design stage; after network deployment, the upper bound of runtime network Â traffic can be obtained by our method.

The remainder of this article is structured as follows. The related works are presented in Section II. Section III presents the system model used in this study and derives EDF scheduling in industrial wireless sensor networks. Section IV defines and formulates the problem. The demand bound function of industrial wireless sensor networks is presented in Section V. Section VI shows the simulation results. Section VII draws the conclusion.

II. RELATED WORKS

Real-time scheduling in industrial wireless sensor networks has gradually caught scholars attention and has been explored in many works. Some researchers consider real-time routing; reference [10] bounds the end-to-end communication delay by enforcing a uniform delivery velocity. Wu et al. [36] present a conflict-aware real-time routing approach, which allows a WSAN to accommodate more real-time flows while meeting their deadlines. References [19], [23]–[25], [37] study the delay analysis in industrial wireless sensor networks by mapping the scheduling of real-time periodic data flows to real-time multiprocessor scheduling. However, these studies are based on source routing. Because graph routing is a reliable approach to handle transmission failures, a few works have begun to focus on graph routing. Saifullah et al. [22] present the first worst-case end-to-end delay analysis for periodic real-time flows under reliable graph routing.

Reference [35] studies the network lifetime maximization problem under graph routing.

In addition, Collotta et al. [8] present a flexible approach in order to improve GTSs assignment and medium access performance based on CSMA/CA-priority; Seno et al. [26] present a technique which is able to grant the feasibility of a set of real-time periodic data exchanges over a wireless network in presence of retransmissions; Reference [38] proposes a NCRF scheme by applying network coding techniques based on a controlled flooding transmission scheme for industrial wireless sensor networks. These works are advanced at improving industrial networks QoS and reliability. However, all these works focus only on single-criticality systems. As described in the previous, priority cannot fully represent the importance. When accidents or errors occur, high-level critical tasks may miss their deadline and spell disasters.

Mixed-criticality systems as a developing tendency of real-time systems lead to new research challenges for industrial wireless sensor networks. The concept of mixed criticality was first proposed in [33] and is quickly becoming an important concept in many systems, especially in Cyber Physical Systems [2], [4], [9], [34]. However, these works mainly focus on uniprocessor or multiprocessor platforms. For communication media, there are a few related works on mixed criticality. [32] proposes a many-core platform that provides mechanisms to integrate applications of different criticality on a single platform. Jin et al. [13] study real-time mixed-criticality communication using the WirelessHART protocol. However, all of these works on mixed-criticality systems are under source routing, which cannot ensure the reliability of the network. Thus, existing system models and solutions cannot be used in our model. It needs to propose a reliability analysis method to solve this issue urgently.

III. SYSTEM MODEL

We consider an industrial network consisting of field devices, one gateway, and one centralized network manager. Our system is proposed in three aspects. We first propose a network model that is abstracted away from mainstream industrial network standards. We then introduce a mixed-criticality system. Finally, we derive EDF scheduling in the industrial network.

A. NETWORK MODEL

In this subsection, we propose our network model. Without loss of generality, our model has the same salient features as WirelessHART [7] and WIA [17], which make it particularly suitable for process industries:

Limiting Network Size: Experiences in process industries have shown the daunting challenges in deploying large-scale WSANs. Typically, 80–100 field devices compose a WirelessHART network with one gateway.

Time Division Multiple Access (TDMA): In industrial wireless sensor networks, time is synchronized and slotted. Because the length of a time slot allows exactly one transmission, TDMA protocols can provide predictable communication latencies and real-time communication.

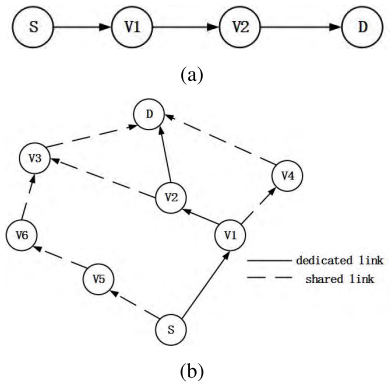


FIGURE 2. Network routing. (a) Source routing, (b) Graph routing.

Route and Spectrum Diversity. To mitigate physical obstacles, broken links, and interference, the messages are routed through multiple paths. Spectrum diversity gives the network access to all 16 channels defined in the IEEE 802.15.4 physical layer and allows per-time slot channel hopping. The combination of spectrum and route diversity allows a packet to be transmitted multiple times, over different channels and different paths, thereby handling the challenges of network dynamics in harsh and variable environments at the cost of redundant transmissions and scheduling complexity [23].

Handling Internal Interference: Industrial networks allow only one transmission in each channel in a time slot across the entire network, thereby avoiding the spatial reuse of channels. Thus, the total number of concurrent transmissions in the entire network at any slot is no greater than the number of available channels [12].

With the above features, the network can be modeled as a graph $G = (V, E, m)$, in which the node set V represents the network devices (all sensor nodes in our model are fixed), E is the set of edges between these devices, and m is the number of channels. Network routing is shown in Fig. 2; our model supports both source routing and graph routing. Source routing is well known in academic research; we will not explore it in this article. Graph routing is a unique feature of industrial wireless sensor networks. In graph routing, a routing graph is a directed list of paths that connect two devices. As shown in Fig. 2(b), graph routing has a primary path and multiple backup paths (The detail of path allocation is out of the range of this paper). This provides redundancy in the route and improves the reliability. As stated in the standard of WirelessHART, for each intermediate node on the primary path, a backup path is generated to handle link or node failure on the primary path. The network manager allocates α dedicated slots, a transmission and $(\alpha-1)$ retransmission on the primary path. A $(\alpha+1)^{th}$ shared slot is allocated on the backup path, usually $\alpha = 2$. In a dedicated slot, one channel only allows one transmission. However, for the case of shared slot, the transmissions having the same receiver can be scheduled in the same slot. The senders that attempt to transmit in a shared

slot contend for the channel using a carrier sense multiple access with collision avoidance (CSMA/CA) scheme [22]. Hence, multiple transmissions can be scheduled in the same channel to contend in a shared slot. For instance, the network manager allocates two dedicated slots for the packet transmits from node S to node $V1$ in Fig. 2(b). After the transmissions on the primary path, a third slot is allocated for the packet transmits from node S to $V5$ as a backup path. When two backup paths intersect at node $V3$, they can avoid collision by CSMA/CA. [30].

It is important to note that the receiver responds with an ACK packet before retransmission and backup; the sender retransmits or sends a backup packet when it does not receive an ACK. Because ACK is a part of the transmission, we do not need to especially analyze the demand of ACK.

B. MIXED-CRITICALITY SYSTEM

A periodic end-to-end communication between a source and a destination is called a *flow*. System switch instruction is a part of control flow. Because we analyze network total demand, we need not distinguish whether a flow is a data flow or a control flow. The total number of flows in the system is n, denoted by $F = \{F_1, F_2, \ldots, F_n\}$. F_i is characterized by $< t_i, d_i, \xi, c_i, \phi_i >$, $1 \leq i \leq n$, where t_i is the period; d_i is the deadline; ξ is the criticality level (we focus on dual-criticality system $\{LO, HI\}$); $\xi = 2$, means the system allocates two slots, one transmission and one retransmission. Our model can be easily extended to systems with an arbitrary number of criticality levels (by increasing the number of retransmissions on the primary path); c_i is the number of hops required to deliver a packet from source to destination. When the system mode switches to high criticality, we denote the total transmission hops of both the primary path and shared paths as C_i; and ϕ_i is the routing path of the flow. Thus, we can describe each flow F_i as follows. F_i periodically generates a packet at its period t_i, and then sends it to the destination before its deadline d_i via the routing path ϕ_i with c_i hops.

At the beginning, messages are transmitted under source routing in low criticality. When an error occurs or the demand changes, the control flow will send a switch instruction, and the system will switch to high-criticality mode. To enhance system reliability, the messages are transmitted under graph routing when the system is running on high-criticality mode. This is an irreversible process; high-criticality mode will never switch back to low-criticality mode (the analytical method of irreversible processes is similar to criticality mode switches from low to high). After the switch, we are not required to meet any deadlines for low-criticality flows, but high-criticality flows may instead execute for up to their high-criticality level characters.

C. EDF SCHEDULING IN INDUSTRIAL NETWORKS

In this subsection, we provide an overview of the earliest deadline first scheduling under industrial wireless sensor networks to analyze system schedulability. EDF scheduling is a commonly adopted policy in practice for real-time

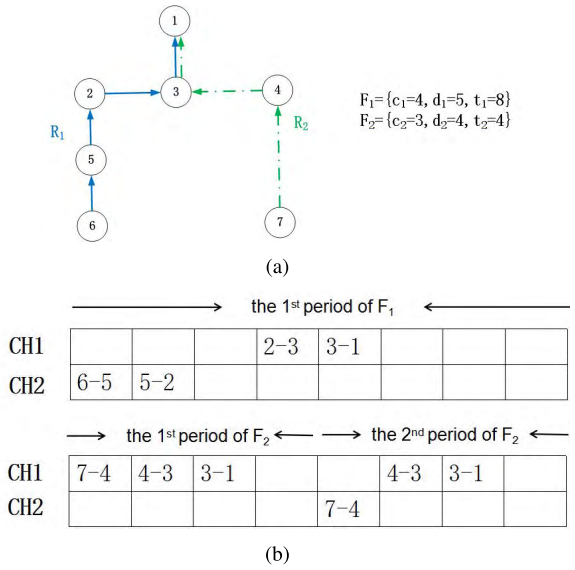


FIGURE 3. An example for EDF scheduling.

CPU scheduling, cyber-physical systems, and industrial networks [29]. In an EDF scheduling policy, each job priority is assigned by its absolute deadline, and the transmission is scheduled based on this priority. Each node in our system is equipped with a half-duplex omnidirectional radio transceiver that can alternate its status between transmitting and receiving. There are two kinds of delay in industrial wireless sensor networks, which can be summarized as follows:

- Channel contention: each channel is assigned to one transmission across the entire network in the same slot.
- Transmission conflicts: whenever two transmissions conflict, the transmission that belongs to the lower-priority job must be delayed by the higher-priority one, regardless of how many channels are available. It is important to note that one node can perform only one operation (receiving or transmitting) in each slot.

In EDF scheduling, the priorityÂ is inversely proportional to its absolute deadline. We explain the operating principle of EDF scheduling by Fig. 3. There are two channels(CH1 and CH2) and flows in this network. At the beginning, the priority of F_2 is higher than F_1 since $d_2 = 4 < d_1 = 5$. Then the controller allocates CH1 for F_2 firstly. The flow with lower priority must be delayed when transmission conflict occurs such as F_1 will be delayed by F_2 at the 3^{th} time slot. At the 5^{th} time slot, the second packet is generated by F_2 with an absolute deadline 8, which is larger than 5. Hence, the priority inversion, and CH1 is allocated to F_1.

Channel contention occurs when high-priority jobs occupy all channels in a time slot; a transmission conflict is generated when several transmissions involve a common node at the same dedicated slot, and a low-priority job is delayed by high-priority ones. However, for the case of shared slots, transmissions with the same receiver can be scheduled in the same slot. When channel contention occurs between backup paths,

the senders on the backup path use a CSMA/CA scheme to contend for the channel, and a network delay will not result in this condition. For a network under graph routing, two paths ϕ_i and ϕ_j involving a common node may conflict in four conditions:

- ϕ_i is a primary path, ϕ_j is a backup path;
- both ϕ_i and ϕ_j are primary paths;
- both ϕ_i and ϕ_j are backup paths;
- ϕ_i is a backup path, ϕ_j is a primary path;

Except for condition 3, the other three conditions may generate transmission conflicts. Consequently, the total delay caused by these conditions depends on how their primary and backup paths intersect in the network.

In real-time system, one task is schedulable when it could be executed completely before its deadline. Hence, the flow could be scheduled when all the packets are generated by the flow could arrive destination before their relative deadlines. Then we define the network schedulability as whether or not all flows in a network are schedulable.

IV. PROBLEM FORMULATION

Given a mixed-criticality industrial network $G = (V, E, m)$, the flow set F and the EDF scheduling algorithm, our objective is to analyze the relationship between the maximum execution demand of the flows and network resource in any time interval such that the schedulability of the flow set can be determined. A successful approach to analyzing the schedulability of real-time workloads is to use demand bound functions [3], [21]. We introduce this concept into industrial wireless sensor networks and propose two definitions as follows:

Definition 1 (Supply Bound Function): A supply bound function sbf(l) is the minimal transmission capacity provided by the network within a time interval of length l.

Definition 2 (Demand-Bound Function): A demand bound function dbf(F_i, l) gives an upper bound on the maximum possible execution demand of flow F_i in any time interval of length l, where demand is calculated as the total amount of required execution time of flows with their whole scheduling windows within the time interval.

There are methods for computing the supply bound function $sbf(l)$ in single-processor systems [20], [27]—for example, a unit-speed, dedicated uniprocessor has $sbf(l) = l$. We say that a supply bound function sbf is of no more than unit speed if

$$sbf(0) = 0 \wedge \forall l, \quad k \geq 0 : \ sbf(l + k) - sbf(l) \leq k. \quad (1)$$

Because each channel can be mapped as one processor, the supply bound function sbf of the industrial network can be bounded as

$$sbf(0) = 0 \wedge \forall l, \quad k \geq 0 : \ sbf(l + k) - sbf(l) \leq Ch \times k, \quad (2)$$

where Ch is the number of channels in the network. Furthermore, as a natural assumption of all proposed virtual resource platforms in the literature, we assume that the supply bound function is piecewise linear in all intervals $[k, k + l]$. In TDM (time division multiple), the system supply bound function can be expressed as

$$sbf(l) = \max(l \mod \Theta - \Theta + \Phi, 0) + \lfloor \frac{l}{\Theta} \rfloor \Phi, \quad (3)$$

where Θ is the period of TDM, and Φ is the length of slots allocated to the transmission.

In different modes, the schedulability of the flow set is determined as follows:

$$\sum_{F_i \in F} dbf_{LO}(F_i, l) \leq sbf_{LO}(l), \quad \forall l \geq 0. \quad (4)$$

$$\sum_{F_i \in HI(F)} dbf_{HI}(F_i, l) \leq sbf_{HI}(l), \quad \forall l \geq 0. \quad (5)$$

Similar to real-time scheduling, the flow set is scheduled when the system is satisfied by equation 4 and 5. However, in contrast to real-time scheduling, there are two kinds of delays in industrial wireless sensor networks, channel contention and transmission conflicts. When a transmission conflict occurs, a high-priority job will influence a low priority job, and thus, the flows are not independent.

Note that transmission conflict is a distinguishing feature in industrial wireless sensor networks that does not exist in conventional real-time processor scheduling problems. To analyze the network demand in any time interval, we must consider the delay caused by transmission conflicts.

Moreover, in mixed-criticality systems, there may be some jobs that are released but not finished at the time of the switch to high-criticality mode; we define these jobs as *carry-over jobs*. We must analyze *carry-over jobs* to tighten the demand bound of the network.

V. DEMAND-BOUND FUNCTION OF INDUSTRIAL NETWORK

In this section, we analyze the network demand bound function for a single-criticality system and mixed-criticality system. For the single-criticality system, we study the demand bound function from channel contention and transmission conflicts. On this basis, we then analyze the delay caused by *carry-over jobs* (the job that is released but not finished at the time of the switch) in the mixed-criticality system. Finally, we study the methods for tightening the network demand bound function.

A. ANALYSIS OF SINGLE-CRITICALITY SYSTEM

In this subsection, we study the demand bound function under a single-criticality system in two steps. First, we formulate network transmission conflict delay with path overlaps; we then analyze the network dbf. To make our study self-contained, we present the results of the state-of-the-art demand bound function for CPU scheduling [9], [31]. Assuming that the flows are executed on a multiprocessor platform, the channel is mapped as a processor. We can obtain the network demand caused by channel contention in any time

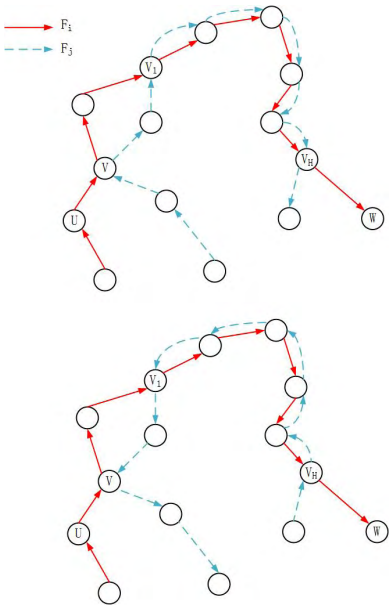


FIGURE 4. An example of transmission delay.

interval l as

$$dbf(l)^{ch} = \frac{1}{m}\sum_{i=1}^{n}\left[\left[\left(\left\lfloor\frac{l-d_i}{t_i}+1\right\rfloor\right)c_i\right]\right]_0. \tag{6}$$

Equation 6 considers only the delay caused by channel contention, denoted as $dbf(l)^{ch}$. The jobs are conflicted when their transmission paths have overlaps. As shown in Fig. 4, the priority of the job in F_i is higher than the one in F_j, so the job in F_j may be delayed by in F_i at nodes V and V_1 to V_h (we assume the network is connected and do not consider the case where the path disconnects).

Transmission conflicts are generated at the path overlaps, and the network requires more resources to solve the transmission conflicts. To obtain $dbf(l)$ of the network, we must first study the relationship between conflict delay and path overlap. However, estimation transmission conflict delay by the length of the overlap is often a pessimistic method. As shown in Fig. 4, the delay is much smaller than the length of the path overlap. To avoid pessimistic estimation, we introduce the result proposed by Saifullah in [25]. The length of the k^{th} path overlap is denoted as $Len_k(ij)$, and its conflict delay is $D_k(ij)$. For the overlap as $V_1 \rightarrow \ldots V_h$, if there exists node $u, w \in V$ such that $u \rightarrow V_1 \rightarrow \ldots V_h \rightarrow w$ is also on F_is route, then $Len_k(ij) = h + 1$. If only u or only w exists, then $Len_k(ij) = h$. If neither u nor v exists, then $Len_k(ij) = h - 1$. In our example, $Len_1(ij) = 2$, $Len_2(ij) = 7$ and $D(ij) = D_1(ij) + D_2(ij)$, which is at most $2 + 3 = 5$. Obviously, $Len_k(ij)$ is the upper bound of $D_k(ij)$, which means $Len_k(ij) \geq D_k(ij)$. For the flow set F, the total delay caused by transmission conflicts Δ is

$$\Delta = \sum_{1\leq i,j\leq n} D_k(ij) \leq \sum_{1\leq i,j\leq n} Len_k(ij). \tag{7}$$

By the Lemma proposed in [25], the estimation of the delay caused by overlap with length of at least 4 can be tightened. We then formulate the total transmission conflicts between F_i and F_j as

$$\Delta(ij) = \sum_{k=1}^{\delta(ij)} Len_k(ij) - \sum_{k'=1}^{\delta'(ij)} (Len_{k'}(ij) - 3), \tag{8}$$

where $\delta(ij)$ is the number of path overlaps, $\delta'(ij)$ is the number of path overlaps with length of at least 4. Because all flows have periodic duty, we denote T as the least common multiple of flow set F (because the period is an integral multiple of 2, T is equal to the longest period among F). Network dbf changes with time interval l while it slides from 0 to T. However, Lemma 2 proposed by Saifullah is scheduled under fixed priority, so the priorities of flows are variable under EDF scheduling. We must analyze whether Saifullah's result is suitable under EDF scheduling. We denote the m^{th} job generated by F_i as F_i^m, and our objective is to estimate the delay caused by transmission conflicts by analyzing the number of conflicts.

Lemma 1: F_i^k and F_j^g are two jobs of flow i and j, when F_i^k and F_j^g ($F_i^k \in hp(F_j^g)$) conflict, the job F_i^k will never be blocked by the job F_j^{g+m}. However, F_i^{k+m} may be blocked by F_j^g.

Proof: At the beginning, the priority of F_i^k is higher than F_j^g, which means $d_i^k < d_j^k$. As Fig. 4 shows, two flows may conflict at V_1, and F_j is delayed by F_i. When F_i^k is forwarded to V_h, two jobs may conflict again. If F_i^k is blocked by F_j^{k+m}, we can obtain $d_i^k > d_j^{g+m}$. Because $d_j^{g+m} > d_j^g$, this contradicts with $d_i^k < d_j^g$. Hence, F_i^k will never be blocked by F_j^{g+m}.

We prove that F_i^{k+m} is blocked by F_j^g through an example. We use the following simple flow set:

Flow	c	d=t
F_1	1	2
F_2	1	3

At the beginning, the priority of F_1^1 is higher than F_2^1, because the absolute deadline is 2 and 3, respectively. At time slot 2, another job is generated by F_1 with the absolute deadline of 2. However, the absolute deadline of F_2^1 is 1, F_1^2 is blocked by F_2^1. Hence, F_i^{k+m} can be blocked by F_j^g. ■

Because a path is a chain of transmissions from source to destination, in considering the conflict delay caused by multiple jobs of F_i on flow F_j, we analyze the number of conflicts for F_i and F_j. Thus, Lemma 2 establishes the upper bound of this value.

Lemma 2: When F_j and F_i conflict, within any time interval of length l, each job of F_j can be blocked no more than $\lceil \frac{l}{t_i} \rceil$ times, and F_j can be blocked by F_j no more than $\lceil \frac{l}{t_j} \rceil$ times.

Proof: Based on Lemma 1, we know that the priority inversion will occur in the process of transmission. If F_i^k is a higher-priority job than F_j^g, the jobs released after F_j^g must

be blocked by F_i^k until F_i^k is finished. If all jobs generated by F_i satisfy $d_i^{k+\lceil \frac{l}{t_i} \rceil} < d_j^g$, where k and g are the first jobs for F_i and F_j, respectively, in l, then there are no more than $\lceil \frac{l}{t_i} \rceil$ jobs of F_i. Beyond that, because there is no transmission conflict, the other jobs of F_j are not blocked by F_i. Hence, F_j can be blocked by F_i no more than $\lceil \frac{l}{t_i} \rceil$ times. The same as F_i, F_i can be blocked by F_j no more than $\lceil \frac{l}{t_j} \rceil$ times. ■

By Lemma 1 and 2, we can estimate the network demand caused by the transmission conflict. Based on equation 6, we obtain the upper bound of $dbf(l)$ as follows:

Theorem 1: In any time interval of length l, the demand bound function under a single-critical network (low-criticality mode) is upper-bounded by

$$dbf_{LO}(l) = \frac{1}{m}\sum_{i=1}^{n}\left[\left[\left(\left\lfloor \frac{l-d_i}{t_i}+1 \right\rfloor\right)c_i\right]\right]_0 + \sum_{1\le i,j\le n}(\Delta(ij)\max\{\lceil\frac{l}{t_i}\rceil,\lceil\frac{l}{t_j}\rceil\}). \quad (9)$$

Proof: Network demand is the upper bound in a time interval of length l, which consists of two parts, channel contention and transmission conflict. The demand of channel contention is bounded by equation 6. For the demand of the transmission conflict, we first analyze each time conflict delay for each two paths by equation 8; the number of conflicts can then be obtained by Lemma 2. We can obtain the network demand of transmission conflict as

$$\sum_{1\le i,j\le n}(\Delta(ij)\max\{\lceil\frac{l}{t_i}\rceil,\lceil\frac{l}{t_j}\rceil\}). \quad (10)$$

Hence, we can obtain the demand bound function under a single-critical network upper-bounded by equation 9. ■

B. ANALYSIS OF MIXED-CRITICALITY SYSTEM

Based on the result proposed in subsection V-A, we extend the idea of a demand bound function to a mixed-criticality system. For illustration purposes, only a dual-criticality system is considered; this means that ξ has only two values, *LO* (low-criticality mode) and *HI* (high-criticality mode). Nevertheless, it can be easily extended to systems with an arbitrary number of criticality modes. We construct three demand bound functions: the demand bound function in low- and high-criticality modes ($dbf_{LO}(l)$ and $dbf_{HI}(l)$) and the demand bound function when system mode switches ($dbf_{LO2HI}(l)$). We analyze $dbf_{HI}(l)$ and $dbf_{LO2HI}(l)$ under graph routing in this subsection.

The system begins from the low-criticality level, and all flows are served and executed as in a single-criticality system. When errors or emergencies occur, the centralized network manager will trigger the switching of the system mode from *LO* to *HI*. In high-criticality mode, the network turns to graph routing, and the flows in the low-criticality level are discarded; only high-criticality flows can be delivered. The job that is active (released, but not finished) from a high-criticality flow at the time of the switch is still running under source routing; n_{HI} is the number of high-criticality flows, and there are no more than n_{HI} *carry-over jobs* that are active at the time of the switch. We define these *carry-over jobs* as new flows $F_{(n_{HI}+1)}, F_{(n_{HI}+2)} \ldots F_{2n_{HI}}$, which have the same characters as the corresponding flows in F except for c and t. For the new flow $F_{p+n_{HI}}$, $c_p > c_{(p+n_{HI})}$, and as an accidental event, $t_{(p+n_{HI})} \gg t_p$.

When the system switches from *LO* to *HI*, the demand of *carry-over jobs* is

$$\frac{1}{m}\sum_{p=1+n_{HI}}^{2n_{HI}} c_p + \sum_{n_{HI}\le p,q\le 2n_{HI}}\Delta(pq). \quad (11)$$

Furthermore, the flows will generate new jobs when the system switches to high-criticality mode. Because each node except the destination on the primary path generates one backup path, the total number of paths for F_p is $c_p + 1$ and the execution time for each backup path c_p^k can be obtained from the network easily. The total execution time of F_i can be denoted as $C_p = c_p + \sum_{k=1}^{c_p} c_p^k$. Therefore, network demand for channel contention under graph routing is

$$dbf_{HI}^{ch}(l) = \frac{2}{m}\sum_{p=1}^{n_{HI}}\left[\left[\left(\left\lfloor \frac{l-d_p}{t_p}+1 \right\rfloor\right)C_p\right]\right]_0. \quad (12)$$

Based on the rules of transmission conflict proposed in section III-C, a transmission conflict between two flows is generated only if there is at least one flow transmission on the primary path. Therefore, we analyze $dbf_{HI}(l)$ by studying the transmission conflict generated on the primary path. For F_p^g and F_q^m, when given $d_p < dq$, F_q^m may be delayed by F_p^g and its backup paths. We denote the path set of F_p and its backup paths as I; each path in I is denoted as p'. The upper bound delay of F_q^m caused by F_p^g is denoted as $\Delta(p'q)$. $\Delta(p'q)$ can be formulated as

$$\Delta(p'q) = \sum_{p'=1}^{c_p+1}\left(\sum_{k=1}^{\delta(p'q)} Len_k(p'q) - \sum_{k'=1}^{\delta'(p'q)}(Len_{k'}(p'q)-3)\right). \quad (13)$$

For the job on the backup path, a transmission delay occurs only when it conflicts with primary paths with high-priority jobs. When we reverse the priority of F_p^g and F_q^m, equation 13 is the upper bound additional demand of F_p^g caused by F_q^m. From the above, the network upper bound demand function under graph routing can be described as

$$dbf_{HI}(l) = \frac{2}{m}\sum_{i=1}^{n_{HI}}\left[\left[\left(\left\lfloor \frac{l-d_i}{t_i}+1 \right\rfloor\right)C_i\right]\right]_0 + \sum_{1\le p,q\le n_{HI}}(\Delta(p'q)\max\{\lceil\frac{l}{t_p}\rceil,\lceil\frac{l}{t_q}\rceil\}). \quad (14)$$

We can then obtain $dbf_{LO2HI}(l)$ as

$$dbf_{LO2HI}(l) = \frac{2}{m}\sum_{p=1}^{n_{HI}}\left(\left[\left(\left\lfloor \frac{l-d_p}{t_p}+1\right\rfloor\right)C_p+\frac{1}{2}c_p\right)\right]_0 + \sum_{1\le p,q\le 2n_{HI}}(\Delta(p'q)\max\{\lceil\frac{l}{t_p}\rceil,\lceil\frac{l}{t_q}\rceil\}). \quad (15)$$

Because transmission on a backup path occurs only when the two previous attempts fail, when the transmission success rate on the primary path satisfies the network packet reception ratio, the sender has no need to send a backup packet. Hence, the network upper bound demand function in this case can be rewritten as

$$dbf_{LO2HI}(l) = \frac{3}{m}\sum_{p=1}^{n_{HI}}\left[\left(\left\lfloor \frac{l-d_p}{t_p}+1\right\rfloor\right)c_p\right]_0 + \sum_{1\le p,q\le 2n_{HI}}(\Delta(pq)\max\{\lceil\frac{l}{t_p}\rceil,\lceil\frac{l}{t_q}\rceil\}). \quad (16)$$

C. TIGHTENING THE DEMAND BOUND FUNCTIONS

A loose demand bound function will lead to a pessimistic estimation of network schedulability. In this subsection, we tighten our demand bound functions by discussing the relationship between two flows and transmission conflict.

In our previous analysis 2, the number of conflict jobs is a conservative estimation as $\max\{\lceil\frac{l}{t_i}\rceil,\lceil\frac{l}{t_j}\rceil\}$. However, this value can be reduced by classifying discussions. We divide this value into the following categories:

- $t_i \le t_j$, and $d_i \le d_j$.
- $t_i \le t_j$, and $d_i \ge d_j$.

When the path of F_i and F_j have overlaps, they may generate transmission conflicts. The delay caused by conflict cannot occur in each slot because the flow does not transmit between d and t. Obviously, when one flow works in its ideal time (between d and t), there is no transmission conflict between F_i and F_j.

Condition 1 is shown in Fig. 5(a); conflict occurs only when both F_i and F_j have job transmissions on the path. For a given l, the number of conflicting jobs can be expressed as

$$\lceil\frac{l}{t_j}\rceil(\lfloor\frac{d_j}{t_i}\rfloor+1). \quad (17)$$

Similarly, we can obtain the number of conflicting jobs in condition 2 as

$$\lceil\frac{l}{t_j}\rceil(\lfloor\frac{d_j}{t_i}\rfloor+1)=2\lceil\frac{l}{t_j}\rceil. \quad (18)$$

We denote the number of conflicts as $Num(ij)$. When we know each flow's routing information, the estimation of $Num(ij)$ can be further precise. By taking the modulus of $\frac{d_j}{t_i}$, we can estimate the maximum length of F_i's residual path as $||\frac{d_j}{t_i}||$. The delay on this residual path is denoted as ψ, and we can obtain ψ as follows:

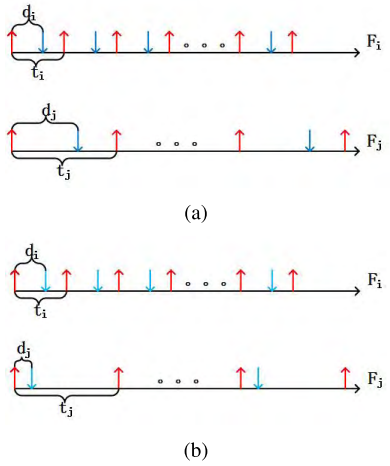


FIGURE 5. Classified discussion. (a) condition 1, (b) condition 2.

- If F_i has an overlap with F_j on this residual path, $\psi = \Delta(||\frac{d_j}{t_i}||)$, where $\Delta(||\frac{d_j}{t_i}||)$ is the delay on the residual path whose length is $||\frac{d_j}{t_i}||$.
- If F_i has no overlap with F_j on this residual path, $\psi = 0$.

The number of conflicts can be expressed as

$$Num(ij) = \lceil\frac{l}{t_j}\rceil(\lfloor\frac{d_j}{t_i}\rfloor+\psi). \quad (19)$$

We can then obtain the system demand bound functions as follows:

Theorem 2: In any time interval of length l, the demand bound function in each mode can be expressed as

$$dbf_{LO}(l) = \frac{1}{m}\sum_{i=1}^{n}\left[\left(\left\lfloor\frac{l-d_i}{t_i}+1\right\rfloor\right)c_i\right]_0 + \sum_{1\le i,j\le n}(\Delta(ij)Num(ij)). \quad (20)$$

$$dbf_{LO2HI}(l) = \frac{2}{m}\sum_{p=1}^{n_{HI}}\left(\left[\left(\left\lfloor\frac{l-d_p}{t_p}+1\right\rfloor\right)C_p+\frac{1}{2}c_p\right)\right]_0 + \sum_{1\le p,q\le 2n_{HI}}(\Delta(p'q)Num(pq)). \quad (21)$$

$$dbf_{HI}(l) = \frac{2}{m}\sum_{p=1}^{n_{HI}}\left(\left[\left(\left\lfloor\frac{l-d_p}{t_p}+1\right\rfloor\right)C_p\right)\right]_0 + \sum_{1\le p,q\le 2n_{HI}}(\Delta(p'q)Num(pq)). \quad (22)$$

The system demand bound function is $dbf(l) = \max\{dbf_{LO}(l), dbf_{LO2HI}(l), dbf_{HI}(l)\}$, *and the system can be scheduled when* $dbf(l)$ *is no less than* $\min\{dbf_{LO}(l), dbf_{LO2HI}(l)\}$.

Proof: The proofs of demand bound functions are similar to in Theorem 1. The difference is that we reduce the number of conflicts by classifying the discussion, and the demand bound functions are tightened. Because there are *carry-over jobs* at the switching time, $dbf_{LO2HI}(l)$ must be greater than $dbf_{HI}(l)$. When the network supply in time interval of length l $sbf(l)$ is larger than $dbf_{LO}(l)$, the system can be scheduled in

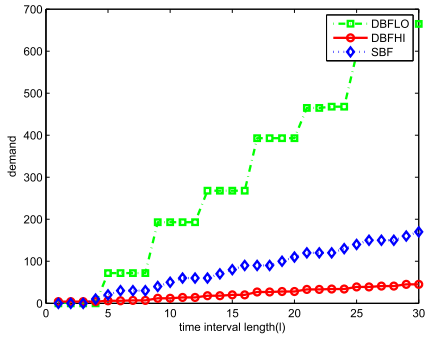


FIGURE 6. Relationship between demand bound functions and supply bound function.

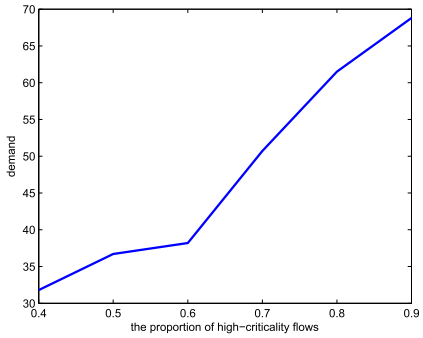


FIGURE 7. Variation tendency of *DBFHI* with the percentage of high-criticality flows.

low-criticality mode; when $dbf_{LO2HI}(l) \leq sbf(l) < dbf_{LO}(l)$, the system can be scheduled in high-criticality mode; when $sbf(l) > \max\{dbf_{LO}(l), dbf_{LO2HI}(l)\}$, the system cannot be scheduled. Hence, the system can be scheduled when $dbf(l)$ is no less than $\min\{dbf_{LO}(l), dbf_{LO2HI}(l)\}$. ■

VI. PERFORMANCE EVALUATION

In this section, we conduct experiments to evaluate the performance of our proposed methods. Our approach is first compared with the simulation result. We then compare our method with the supply/demand bound function analysis without tightening.

To illustrate the applicability of our method, for each parameter configuration, several test cases are generated randomly. For each test case, the network gateway is placed at the center of playground area A, and the other nodes are deployed randomly around the gateway. According to the suggestion in [6], given the transmitting range $d = 40m$, the number of nodes n and the playground area A should satisfy

$$\frac{n}{A} = \frac{2\pi}{d^2\sqrt{27}}. \tag{23}$$

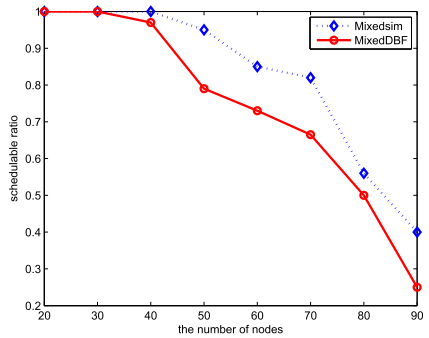
If two nodes can communicate with each other, which means that the distance between two nodes is less than d, they are adjacent nodes. By repeatedly connecting the nearest node from the source node to gateway, network topology can be obtained. If some source nodes cannot connect to the gateway, their locations are generated randomly again.

Our simulations use the utilization u to control the workload of the entire network. To make flow sets available, we specify the network utilization $U = \sum u_i (U < 1)$, and the UUniFast algorithm [5] is used to generate each flow's utilization u_i ($u_i = \frac{c_i}{t_i}$). The result generated by the UUniFast algorithm follows a uniform distribution and is neither pessimistic nor optimistic for the analysis [5].

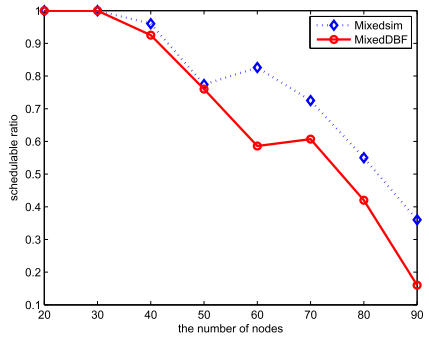
Fig. 6 is an example of the relationship between the demand bound function in different criticality modes and the supply bound function. In this example, according to the actual situation, we set the number of nodes as $n = 70$ and the number of flows as $F = 20$. At the beginning, with the network running in low-criticality mode, the demand is zero. At time slot 5, *DBFLO* is 72, which is larger than the upper bound of network supply; the system then switches to high-criticality mode. Considering *carry-over jobs*, we can calculate the demand in high-criticality mode form time slot 5. Because the network demand is less than the supply, this example is a stable system. Furthermore, Fig. 6 reveals that the demand bound functions are stepwise increasing. This is because $dbf(l)$ is the network demand over a period of time. When a job has enough time slots to transmit (e.g., a job is just released), its demand is zero and does not require immediate execution. With the decrease of the remaining time, the job becomes urgent. When the remaining time for the job is c, the job must be forwarded immediately; otherwise, it will miss the deadline. The job demand is then changed to the number of hops c.

Fig. 7 is the variation tendency of *DBFHI* with the proportion of high-criticality flows. Because changing the proportion of n_{HI} does not affect system demand in low-criticality mode, Fig. 7 shows the network demand only in high-criticality mode. Obviously, the network demand is increasing with increasing proportion of high-criticality flows. At the beginning (0.4–0.6), the network demand increases slowly. From 0.7 − −0.9, the demand of the network increases rapidly. This is because more flows in high-criticality mode generate more transmission conflicts in conditions 1, 2, and 4. The network needs more resources to ensure that the job meets its deadline. This phenomenon is enhanced severely with increasingly P.

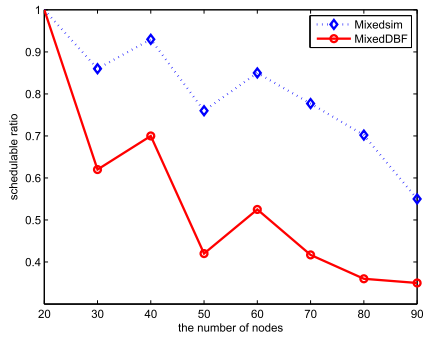
To analyze the correctness of our method, we compare the network schedulability ratio between the simulation result (denoted as MixedSim) and our method (denoted as MixedEDF) in Fig. 8. For each point in the figures, more than 100 test cases are randomly generated. From the figures, we can know that our algorithm can accurately evaluate the network schedulability ratio regardless of which parameters are used. Because we pessimistically estimate transmission conflicts to guarantee our methods reliability, the evaluation value of the network demand bound is larger than the actual demand. In Fig .8(a) and Fig. 8(b), the proportions of high-criticality flows are $P = 0.4$ and $P = 0.5$, respectively. With the increasing of nodes, the network schedulability ratio declines in both situations. However, the schedulability ratio in Fig. 8(b) falls faster than in Fig .8(a). This is because the



(a)



(b)



(c)

FIGURE 8. Relationship between schedulability ratio and the number of nodes. (a) $U = 0.5, P = 0.4$, (b) $U = 0.5, P = 0.5$, (c) $U = 0.6, P = 0.4$.

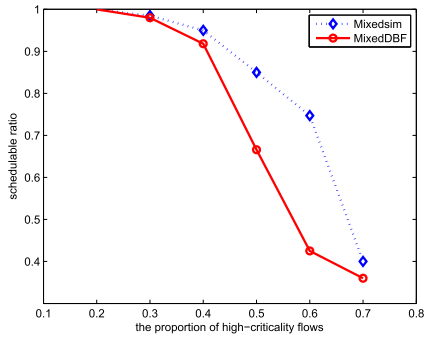
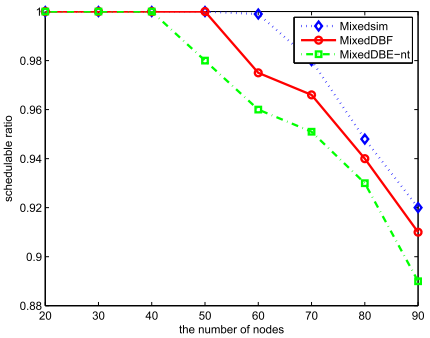
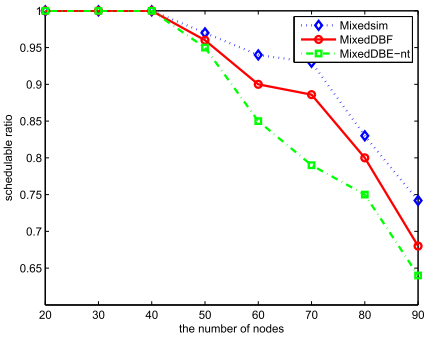


FIGURE 9. Relationship between schedulability ratio and the proportion of high-criticality flows.

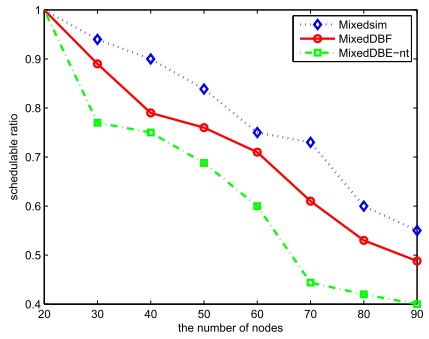
network generates more transmission conflicts when increasing the number of high-criticality flows. Note that compared



(a)



(b)



(c)

FIGURE 10. Schedulability comparison among MixedSim, MixedDBF, MixedDBF-nt. (a) $U = 0.4, P = 0.2$, (b) $U = 0.5, P = 0.2$, (c) $U = 0.4, P = 0.6$.

with Fig .8(a), Fig .8(c) has 0.1 additional utilization, so the spacing between the simulation curve and analysis curve is expanded. Although there are fluctuations between 30 to 60, our method can always bound the schedulable ratio (the fluctuations are caused by the randomly generated network environment). Because the two figures generate test cases according to the respective utilization, their test cases are different. When network utilization increases, the number of hops from source to destination increases. This increases the number of potential conflicts. The estimation result then becomes more pessimistic.

Fig. 9 is the relationship between the schedulability ratio and the proportion of high-criticality flows. It is easy to understand that the schedulability ratio declines with the

increasing proportion of high-criticality flows. However, the spacing between the two curves changes with P (smallâĂ"bigâĂ"small). This is because our method should consider the transmission conflicts in all situations to ensure reliability. At the beginning, there are only a few conflicts in high-criticality mode. With increasing high-criticality flows, the strict estimation considers each path overlap as a transmission conflict, which leads to a larger spacing between two curves. When $P = 0.7$, the number of conflicts increases in MixedSim, which reduces the schedulability ratio, and then the difference becomes small.

We illustrate the advantage of MixedDBF by comparing it with the supply/demand bound function analysis without tightening (denoted as MixedDBF-nt) in Fig. 10. Obviously, MixedDBF is better than MixedDBF-nt regardless of the conditions. With increasing network utilization or proportion of high-criticality flows, the error of MixedDBF-nt grows faster than MixedDBF. The reason is that both increasing network utilization and the number of high-criticality flows will increase the number of path overlaps. MixedDBF tightens the delay caused by the transmission conflict by equation 19. With increasing overlaps, the effect of equation 19 will be better. Hence, the error of MixedDBF-nt grows faster than MixedDBF.

VII. CONCLUSION

Reliability and real time are the most important characteristics of industrial wireless sensor networks. Standards such as WirelessHART adopt a reliable graph routing to enhance network reliability. However, there are trade-offs. Graph routing introduces substantial challenges in analyzing the schedulability of real-time flows. Too much transmission load will increase conflicts and reduce system performance. Disaster may happen when critical tasks miss their deadlines in this situation. Hence, we propose a mixed-criticality industrial wireless sensor networks to solve this issue. To our best knowledge, this is the first study for mixed criticality under both source and graph routings. By introducing the concept of mixed criticality in industrial wireless sensor networks, we propose a novel network model that can switch routing based on system criticality mode. When errors or accidents occur, system switches to high criticality mode and low-level critical tasks are abandoned. Then, we analyze the demand bound of mixed-criticality industrial wireless sensor networks and formulating network demand bounds in each criticality mode. In addition, we tighten the demand bound by analyzing carry-over jobs and classifying the number of conflicts to improve our theory's accuracy. The system can be scheduled when it satisfies our method even though in the worst case. Simulations based on random network topologies demonstrate that our method can estimate network schedulability efficiently.

Future work will deal with improve the reliability by optimizing the scheduling policy of mixed criticality industrial wireless sensor networks.

REFERENCES

[1] *ISA 100*, accessed on Aug. 12, 2015. [Online]. Available: http://www.isa.org/isa100

[2] S. Baruah, H. Li, and L. Stougie, "Towards the design of certifiable mixed-criticality systems," in *Proc. 16th IEEE Real-Time Embedded Technol. Appl. Symp. (RTAS)*, Apr. 2010, pp. 13–22.

[3] S. K. Baruah, A. K. Mok, and L. E. Rosier, "Preemptively scheduling hard-real-time sporadic tasks on one processor," in *Proc. 11th Real-Time Syst. Symp.*, Dec. 1990, pp. 182–190.

[4] S. K. Baruah *et al.*, "Scheduling real-time mixed-criticality jobs," *IEEE Trans. Comput.*, vol. 61, no. 8, pp. 1140–1152, Aug. 2012.

[5] E. Bini and G. C. Buttazzo, "Measuring the performance of schedulability tests," *Real-Time Syst.*, vol. 30, nos. 1–2, pp. 129–154, 2005.

[6] T. Camilo, J. S. Silva, A. Rodrigues, and F. Boavida, "Gensen: A topology generator for real wireless sensor networks deployment," in *Proc. IFIP Int. Workshop Softw. Technol. Embedded Ubiquitous Syst. (SEUS)*, Santorini Island, Greece, May 2007, pp. 436–445.

[7] D. Chen, M. Nixon, and A. Mok, *Why WirelessHART*. Germany: Springer, 2010.

[8] M. Collotta, L. Gentile, G. Pau, and G. Scatá, "Flexible IEEE 802.15.4 deadline-aware scheduling for DPCSs using priority-based CSMA-CA," *Comput. Ind.*, vol. 65, no. 8, pp. 1181–1192, 2014.

[9] P. Ekberg and W. Yi, "Outstanding paper award: Bounding and shaping the demand of mixed-criticality sporadic tasks," in *Proc. 24th Euromicro Conf. Real-Time Syst. (ECRTS)*, 2012, pp. 135–144.

[10] T. He, J. A. Stankovic, C. Lu, and T. Abdelzaher, "Speed: A stateless protocol for real-time communication in sensor networks," in *Proc. 23rd Int. Conf. Distrib. Comput. Syst.*, 2003, pp. 46–55.

[11] Z. He, W. Zhang, H. Wang, and W. Huang, "Joint routing and scheduling optimization in industrial wireless networks using an extremal dynamics algorithm," *Inf. Control*, vol. 43, no. 2, pp. 152–158, 2014.

[12] A. Herms, S. Schemmer, and G. Lukas, "Real-time mesh networks for industrial automation," in *Proc. SPS/IPC/DRIVES, Elektrische Automatisierung, Syst. Komponenten*, vol. 7. 2007, pp. 1–18.

[13] X. Jin, J. Wang, and P. Zeng, "End-to-end delay analysis for mixed-criticality WirelessHART networks," *IEEE/CAA J. Autom. Sinica*, vol. 2, no. 3, pp. 282–289, Jul. 2015.

[14] O. Kaiwartya *et al.*, "T-MQM: Testbed-based multi-metric quality measurement of sensor deployment for precision agriculture—A case study," *IEEE Sensors J.*, vol. 16, no. 23, pp. 8649–8664, Dec. 2016.

[15] L. Kong, D. Zhang, Z. He, Q. Xiang, J. Wan, and M. Tao, "Embracing big data with compressive sensing: A green approach in industrial wireless networks," *IEEE Commun. Mag.*, vol. 54, no. 10, pp. 53–59, Oct. 2016.

[16] L. Kong, C. Yuan, H. Xiao, and S. Xiao, "Stochastic optimization model and solution algorithm for blending procedure of process industrial," *Inf. Control*, vol. 45, no. 1, pp. 40–44, 2016.

[17] W. Liang, X. Zhang, Y. Xiao, F. Wang, P. Zeng, and H. Yu, "Survey and experiments of wia-pa specification of industrial wireless network," *Wireless Commun. Mobile Comput.*, vol. 11, no. 8, pp. 1197–1212, 2011.

[18] G. Lipari, "Earliest deadline first, Edf.texSistemi in tempo real," Scuola Superiore Sant'Anna, Pisa, Italy, Jun. 2005.

[19] C. Lu *et al.*, "Real-time wireless sensor-actuator networks for industrial cyber-physical systems," *Proc. IEEE*, vol. 104, no. 5, pp. 1013–1024, May 2016.

[20] A. K. Mok, X. Feng, and D. Chen, "Resource partition for real-time systems," in *Proc. 7th IEEE Real-Time Technol. Appl. Symp.*, May/Jun. 2001, pp. 75–84.

[21] J. Rox and R. Ernst, "Compositional performance analysis with improved analysis techniques for obtaining viable end-to-end latencies in distributed embedded systems," *Int. J. Softw. Tools Technol. Transf.*, vol. 15, no. 3, pp. 171–187, 2013.

[22] A. Saifullah *et al.*, "Schedulability analysis under graph routing in WirelessHART networks," in *Proc. RTSS*, 2015, pp. 165–174.

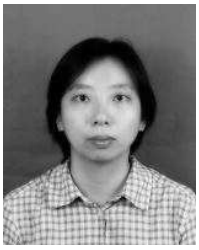
[23] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "Real-time scheduling for WirelessHART networks," in *Proc. IEEE 31st Real-Time Syst. Symp. (RTSS)*, Nov./Dec. 2010, pp. 150–159.

[24] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "End-to-end delay analysis for fixed priority scheduling in WirelessHART networks," in *Proc. 17th IEEE Real-Time Embedded Technol. Appl. Symp. (RTAS)*, Apr. 2011, pp. 13–22.

[25] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "End-to-end communication delay analysis in industrial wireless networks," *IEEE Trans. Comput.*, vol. 64, no. 5, pp. 1361–1374, May 2015.

[26] L. Seno, A. Valenzano, and C. Zunino, "A dynamic bandwidth reassignment technique for improving QoS in EDF-based industrial wireless networks," in *Proc. IEEE 13th Int. Conf. Ind. Inform. (INDIN)*, Jul. 2015, pp. 892–899.
[27] I. Shin and I. Lee, "Periodic resource model for compositional real-time guarantees," in *Proc. 24th IEEE Real-Time Syst. Symp. (RTSS)*, Dec. 2003, pp. 2–13.
[28] J. Song *et al.*, "WirelessHART: Applying wireless technology in real-time industrial process control," in *Proc. IEEE Real-Time Embedded Technol. Appl. Symp. (RTAS)*, Apr. 2008, pp. 377–386.
[29] M. Spuri, "Analysis of deadline scheduled real-time systems," Inst. Nat. Recherche Inform. Autom., Tech. Rep. 2772, 1996.
[30] *Industrial Communication Networks—Wireless Communication Network and Communication Profiles—WirelessHART*, IEC Standard 62591:2010, Int. Electrotech. Commission, 2010.
[31] M. Tillenius, E. Larsson, R. M. Badia, and X. Martorell, "Resource-aware task scheduling," *ACM Trans. Embedded Comput. Syst.*, vol. 14, no. 1, 2015, Art. no. 5.
[32] S. Tobuschat, P. Axer, R. Ernst, and J. Diemer, "IDAMC: A NoC for mixed criticality systems," in *Proc. IEEE 19th Int. Conf. Embedded Real-Time Comput. Syst. Appl. (RTCSA)*, Aug. 2013, pp. 149–156.
[33] S. Vestal, "Preemptive scheduling of multi-criticality systems with varying degrees of execution time assurance," in *Proc. 28th IEEE Int. Real-Time Syst. Symp. (RTSS)*, Dec. 2007, pp. 239–243.
[34] M. Wolf, *High-Performance Embedded Computing: Applications in Cyber-Physical Systems and Mobile Computing*. The Netherlands: Newnes, 2014.
[35] C. Wu *et al.*, "Maximizing network lifetime of wirelesshart networks under graph routing," in *Proc. IEEE 1st Int. Conf. Internet-Things Design Implement. (IoTDI)*, Apr. 2016, pp. 176–186.
[36] C. Wu, D. Gunatilaka, M. Sha, and C. Lu, "Conflict-aware real-time routing for industrial wireless sensor-actuator networks," *Caries Res.*, vol. 18, no. 5, pp. 401–407, 2015.
[37] C. Wu, M. Sha, D. Gunatilaka, A. Saifullah, C. Lu, and Y. Chen, "Analysis of EDF scheduling for wireless sensor-actuator networks," in *Proc. IEEE 22nd Int. Symp. Quality Service (IWQoS)*, May 2014, pp. 31–40.
[38] K. Yu, J. Yue, Z. Lin, J. Åkerberg, and M. Björkman, "Achieving reliable and efficient transmission by using network coding solution in industrial wireless sensor networks," in *Proc. IEEE 25th Int. Symp. Ind. Electron. (ISIE)*, Jun. 2016, pp. 1162–1167.

CHANGQING XIA received the Ph.D. degree from Northeastern University, China, in 2015. He is currently an Assistant Professor with the Shenyang Institute of Automation, Chinese Academy of Sciences. His research interests include wireless sensor networks and real-time systems, especially the real-time scheduling algorithms, and smart energy systems.



XI JIN received the Ph.D. degree from Northeastern University, China, in 2013. She is currently an Associate Professor with the Shenyang Institute of Automation, Chinese Academy of Sciences. Her research interests include wireless sensor networks and real-time systems, especially the real-time scheduling algorithms, and worst case end-to-end delay analysis.

LINGHE KONG received the B.E. degree from Xidian University in 2005, the master's degree from TELECOM SudParis in 2007, and the Ph.D. degree from Shanghai Jiao Tong University in 2012. He was a Post-Doctoral Researcher with Columbia University, with McGill University, and with the Singapore University of Technology and Design. He is currently an Associate Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University. His research interests include wireless communication, sensor networks, mobile computing, Internet of things, and smart energy systems.

PENG ZENG received the Ph.D. degree from the Shenyang Institute of Automation, Chinese Academy of Sciences. He is currently a Professor with the Shenyang Institute of Automation, Chinese Academy of Sciences. His research interests include industrial communication and wireless sensor networks.