

Research Article

Latency Improvement Strategies for Reliability-Aware Scheduling in Industrial Wireless Sensor Networks

Felix Dobsław, Tingting Zhang, and Mikael Gidlund

Computer Science Department, Mid Sweden University, Östersund and Sundsvall, Sweden

Correspondence should be addressed to Felix Dobsław; felix.dobsław@miun.se

Received 9 December 2014; Revised 27 April 2015; Accepted 2 May 2015

Academic Editor: Anfeng Liu

Copyright © 2015 Felix Dobsław et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose novel strategies for end-to-end reliability-aware scheduling in Industrial Wireless Sensor Networks (IWSNs). Because of stringent reliability requirements in industrial applications where missed packets may have disastrous or lethal consequences, all IWSN communication standards are based on Time Division Multiple Access (TDMA), allowing for deterministic channel access on the MAC layer. We therefore extend an existing generic and scalable reliability-aware scheduling approach by the name of SchedEx. SchedEx has proven to quickly produce TDMA schedules that guarantee a user-defined end-to-end reliability level ρ for all multihop communication in a WSN. Moreover, SchedEx executes orders of magnitude faster than recent algorithms in the literature while producing schedules with competitive latencies. We generalize the original problem formulation from single-channel to multichannel scheduling and propose a scalable integration into the existing SchedEx approach. We further introduce a novel optimal bound that produces TDMA schedules with latencies around 20% shorter than the original SchedEx algorithm. Combining the novel strategies with multiple sinks, multiple channels, and the introduced optimal bound, we could through simulations verify latency improvements by almost an order of magnitude, reducing the TDMA superframe execution times from tens of seconds to seconds only, which allows for a utilization of SchedEx for many time-critical control applications.

1. Introduction

The advantages of wireless technologies with respect to flexibility and cost compared to wired solutions together with the introduction of industrial wireless communication standards have led to a growing deployment of Industrial Wireless Sensor Networks (IWSNs) [1]. For an adaptation of WSN for control applications in industrial automation, though, the communication protocol stack must in a scalable way offer end-to-end quality of service (QoS) guarantees which is an acknowledged research challenge lacking a general enough solution [2, 3]. Further, actuator support in the problem formulation, as well as coexistence of interfering networks, is among the open issues that require attention from the research community [3].

IWSN provides an opportunity for flexible and cost-effective sensor networks but introduces difficult challenges. In scenarios with stringent end-to-end QoS requirements, wireless is not trivially implemented, mostly because WSNs suffer from a shared and open communication medium. QoS

features, such as end-to-end latency or reliability, can be addressed on different communication layers. For instance, an improvement in terms of end-to-end reliability can be achieved by a change of route on the network layer, or a rescheduling on the MAC layer. One prominent approach to achieve end-to-end QoS guarantees is to apply contention-free medium access protocols, such as time-division medium access (TDMA) [4]. TDMA assigns dedicated transmission rights to the sensors in a repeating schedule. Energy-efficiency due to duty-cycling and a guaranteed access to the communication medium are advantages of TDMA, as opposed to the best-effort strategies based on carrier sensing [5, 6]. This feature is especially relevant in densely populated sensor networks, with stringent QoS demands, such as for automation control in factories.

WirelessHART [7], the most widespread standard for IWSN, supports an IEEE 802.15.4e amendment for the MAC layer using a variant of TDMA on top of IEEE 802.15.4 2006/2012 PHY. WirelessHART does not dictate the choice of scheduling algorithm: it is the vendor who decides

the implementation. Finding minimal size TDMA schedules is an NP-hard problem to solve, both in the single-channel case where concurrent noninterfering transmissions are allowed (*slot-reuse* [8]) and in the case of multichannel use (e.g., WirelessHART), where concurrent transmissions in dedicated slots may only be scheduled on different channels [9]. Combining multichannel use and slot-reuse is desired in order to further reduce the achievable schedule sizes but brings along a large complexity increase in the formal problem formulation. In order to address end-to-end QoS requirements, the common assumption for IWSN is that all network link quality information is collected at the *network manager* which stands responsible for scheduling, routing, and the distribution of control messages on the downlink to the sensors for both topology deployment and adjustment. The centralized collection of global control information allows for network wide optimal choices and QoS optimization with respect to, for example, fairness or jitter, which would otherwise not be possible in a decentralized setting.

The focus of this paper is on scalable reliability-aware scheduling. A scheduling algorithm is here defined as *reliability-aware*, if it guarantees a user-defined statistic within-network end-to-end reliability $\underline{\rho} \in [0, 1]$ that all packets created within a superframe arrive at their destination at the end of a predefined time interval. As opposed to existing work concerning reliable packet delivery [10, 11], a maximum latency can be guaranteed due to the integration with TDMA and controlled redundancy in terms of repeated transmissions. The proposed metrics for assessment are therefore the minimization of the schedule frame to minimize the maximum guaranteed latency under the guarantee of end-to-end reliability $\underline{\rho}$. We significantly extend the work in [12] with the following contributions.

- (i) The performance of the recent reliability-aware scheduling algorithm SchedEx from [12] is here improved by introducing an optimal bound that reduces latencies for the investigated convergecast topologies by on average 20%, while guaranteeing the same end-to-end reliability levels as in the original paper. A proof of correctness with respect to end-to-end reliability is given too.
- (ii) The results of the scalable extension show that the schedules are calculated about 20% faster than those in the original paper and that the performance in terms of latency outperforms those created by a recent but poorly scaling approach from [13] in all tested topologies.
- (iii) To the best of our knowledge, this is the first paper addressing scalable TDMA scheduling combining slot-reuse, multiple channels, and reliability-aware scheduling in combination. We provide numerous simulation results that verify how even for stringent end-to-end reliabilities of 99.999% all sensor readings can be collected within seconds using schedules that are calculated within microseconds, assuming a variant of WirelessHART that supports slot-reuse.

- (iv) The paper makes a case for the simple extendability of SchedEx for other features such as multichannel use and multisink deployment.

Where convergecast with a single sink does not suffice, SchedEx is here evaluated with different sink deployment plans. Our simulations reveal the latency gain that can be expected for the given topologies using reliability-aware scheduling, highly relevant for mission-critical applications.

The structure of this paper is as follows. Section 2 presents the relevant research in the area. In Section 3 the network model is described, and Section 4 defines the formal problem formulation. The SchedEx algorithm is explained in Section 5. Section 6 introduces the improvements and extensions. The simulations are explained in Section 7, and the results are presented in Section 8. A discussion and concluding remarks follow in Sections 9 and 10, respectively.

2. Related Work

A lack of a generic framework that addresses the end-to-end QoS concerns for mission-critical applications over WSN has been reported in [2] with regards to the MAC layer, and more generally in [3]. Many noteworthy contributions to QoS-aware scheduling have been made [14–20], but since the requirements in industrial automation differ heavily from traditional applications for WSN, alternative approaches are required.

The work in [17] investigates multichannel transmissions with dynamic channel assignment to reduce interference. Network lifetime maximization has been investigated under the consideration of energy harvesting [20], applying cross-layer optimization [19], and the use of redundant actors that support the network in regions where sensor availability drops with time [18]. However, contention-based approaches, such as [17–20], are not suitable to guarantee end-to-end reliability, because of their best-effort nature.

TDMA-based MAC protocols are the most common means to ensure reliability and latency bounds for WSNs (e.g., [8, 12, 21, 22]). TDMA scheduling is an NP-hard combinatorial problem in its most basic static version without the consideration of time synchronization or reliability guarantees [8]. Finding optimal or even initial branch-and-cut solutions using traditional Integer Programming can take weeks or months for moderately large topologies of sizes around 50 nodes without even considering stringent QoS constraints.

In [8], the authors prove the schedule minimization problem with guaranteed packet delivery to be NP-hard and propose two scalable scheduling algorithms to solve it. In [22], the authors show how their SAS-TDMA outperforms the algorithms from [8] under the consideration of a harsh dynamic industrial environment. However, SAS-TDMA does not come with any guarantees in terms of reliability. The TDMA-based multichannel scheduling approaches introduced in [23, 24] address the scheduling problem assuming static network topologies without the consideration of slot-reuse or reliability guarantees. Offline dimensioning approaches, such as GinMAC [25] and Burst [26], address

the problem for industrial networks by maximizing end-to-end reliability under latency constraints. However, they cannot be considered for large or dynamic networks because of their time-consuming execution. The authors in [13] introduce a two-phase scheduling strategy that extends existing scheduling algorithms and maximizes end-to-end reliability under a fixed latency constraint, outperforming the algorithms introduced in [8]. They further investigate redundancy in form of multipath routing in both theory and simulations, and their results show the superiority of single-path routing, not even considering communication overhead in their comparison. The SchedEx scheduling approach is introduced in [12]. SchedEx produces schedules that guarantee a given end-to-end reliability while cooptimizing latency. Simulations show that SchedEx is an order of magnitude faster than the strategy from [13]. So far, SchedEx has only been investigated on single sink topologies and a single communication channel. In this paper the SchedEx approach is further improved with respect to execution time and end-to-end latency performance. Simulations verify its suitability even for multichannel and multisink scenarios, compared to existing work.

3. Model Assumptions

We base our network model, optimization problem formulation, and simulation setup on the work in [12], extending it for multiple channel use in order to allow for an easy comparison between the approaches. Multiple sinks are already supported but have not yet been evaluated. A WSN is defined as a digraph $G = \{V, E\}$ with the set of sensors $V = \mathcal{T} \cup \mathcal{S}$, where \mathcal{T} are the transceivers, \mathcal{S} sinks, and $\mathcal{T} \cap \mathcal{S} = \emptyset$. The successful arrival of a packet is modeled as the node-to-node packet arrival rate that includes the successful reception of an acknowledgement (ACK) on the sender side before the end of the time-slot has been reached. $E = \{(t, p) \mid q_{tp} > 0\}$ is the set of directed links from any transceiver $t \in \mathcal{T}$ to any sensor $p \in V$. Q is the link quality matrix of size $|\mathcal{T}| \times |V|$. Node-to-node packet reception rates are expressed as q_{tp} . The state of the network is represented by a buffer $b = (b_1, \dots, b_{|\mathcal{T}|})$, where b_t is the amount of packets waiting for transmission at sensor t . $b = \vec{x}$ signifies that each sensor has equally x packets in the local buffer. All sensors operate in half-duplex mode and time synchronization between the sensors is assumed to be guaranteed by the lower layers of the protocol stack. Multihop routing is allowed. Packets are produced at the beginning of a schedule frame F and the common deadline to be held is the final slot in the schedule, after which all sensors have to have succeeded to transmit all their packets. Packets that could not be transmitted in F are removed from the buffer in order to preclude contention problems. Assuming stochastic independence among transmissions, any two transmissions from t to p and t' to p' succeed with a probability

$$P_{\text{success}}(tp, t'p') = q_{tp}q_{t'p'}. \quad (1)$$

For reasons of simplicity, topology changes are not considered and all channels of a link have the same link quality q_{tp} .

4. Problem Formulation

Let $o : \mathcal{F} \rightarrow \mathbb{R}$ be an objective function that assigns each configuration of the network $x \in \mathcal{F}$ a quality index.

In this paper, the end-to-end delay in terms of schedule frame size for a given end-to-end reliability demand $\underline{\rho}$ shall be minimized as in

$$\min_x o(x) = |F|, \quad \forall x = (R, F) \in \mathcal{F} \quad (2)$$

$$\text{subject to } c_0: \underline{\rho} \leq \rho \quad (3)$$

$$c_1: r_{tt} = 0 \quad (4)$$

$$c_2: \sum_{p \in V} r_{tp} \leq 1 \quad (5)$$

$$c_3: r_{tp} = 1 \implies (t, p) \in E \quad (6)$$

$$c_4: f_{st} = 1 \implies f_{sp}q_{pt} = 0 \quad (7)$$

$$c_5: f_{st}f_{st'} = 1 \implies q_{tp}q_{t'p} = 0 \quad \forall \quad (8)$$

$$\sum_{t \in \mathcal{T}} f_{st}r_{tp} = 0 \quad \forall \quad (9)$$

$$\text{ch}_{ts} \neq \text{ch}_{t's} \wedge r_{tp} + r_{t'p} \leq 1. \quad (10)$$

End-to-end reliability $\rho \in [0, 1]$, schedule frame F with its transmission rights $f_{st} \in \{0, 1\}$, and channel assignment $\text{ch}_{ts} \in \{1, \dots, \text{CH}_{\text{MAX}}\}$ are the variables to be identified. All constraints c_i must hold true for x in order to be considered a feasible solution $x \in \mathcal{F}$. It follows an explanation of the equations.

A *schedule frame* F is a binary matrix of size $m \times |\mathcal{T}|$ that with $f_{st} = 1$ assigns transmitter $t \in \mathcal{T}$ transmission allowance at discretized time k in *time-slot* $s = k \bmod m$. F is continuously repeated in order to ensure that the scheduled transceivers can transmit periodically. In (2), $|\cdot|$ signifies the size of the frame, m . $\text{SA}(R) = F$ describes the deterministic scheduling algorithm SA that creates a schedule frame F given a single-path routing table R . $\underline{\rho}$ in (3) is the required user-defined end-to-end reliability and ρ the actual end-to-end reliability of the network.

The formal routing constraints (4)–(6) are expressed over a binary routing table R of size $|\mathcal{T}| \times |V|$ that by $r_{tp} = 1$ assigns each transceiver in row t one recipient, its so called *parent*, p . A *successful* routing table that fulfills all constraints is here expected to be supplied by the network layer.

A schedule F is called *valid* or *collision-free* if no receiving sensor is disturbed by a second concurrent transmission (7)–(8) (either no sensor exists that hears two concurrent transmissions or that sensor is not interested in either transmission, or both send via different channels while not sharing the same parent). According to the constraints, spatial-reuse within a slot is allowed if no conflicts arise and transceivers may be scheduled in multiple slots of the same superframe. Constraint (10) for channel assignment has been added in this paper. Channel assignment is realized by $\text{ch}_{ts} \in \{1, \dots, \text{CH}_{\text{MAX}}\}$, the channel over which transceiver t transmits in time-slot s of frame F .

Input: Scheduled Transmitter $t \in \mathcal{T}$, Sinks \mathcal{S} , Buffer State b , Counter Vector c , Repetition Vector τ

- (1) **if** $b_t > 0$ **then**
- (2) c_t-- // count down attempts for transmitter t
- (3) **if** $c_t = 0$ **then**
- (4) b_t-- // remove packet from transmitter buffer
- (5) **if** $R_t \notin \mathcal{S}$ **then**
- (6) $b_{R_t}++$ // add packet to receiver buffer
- (7) $c_t \leftarrow \tau_t$ // update counter

SchedEx, the slot-based scheduling algorithm extension improved in this paper by a tighter bound introduced further below.

ALGORITHM 1: Update Packet Buffers ($t, \mathcal{S}, b, c, \tau$).

5. SchedEx

In [12], the authors compare the two-phase schedule reliability enhancer from [13] and a novel scheduling algorithm extension for reliability-awareness by name SchedEx, with respect to performance and runtime. They show that the schedule enhancer does not scale in the size of the network by both complexity analysis and simulations. For WSNs in industrial applications, the deployed sensors are not capable of running long processes to identify better schedules, because of battery drainage and restricted computational power. SchedEx shows to scale well in the size of the network and guarantees a reliability lower bound $\underline{\rho}$ instead of calculating the exact reliability as done in [13]. Link quality varies over time, and calculating an exact reliability is only of theoretical interest, whereas a swiftly produced schedule that guarantees a lower bound is of practical importance.

The two-phase schedule enhancer from [13] produced schedules that are on average between 5 and 20% shorter than those for SchedEx, with a cost between 4000 and 12000% runtime for networks with node-to-node reliability levels produced by the Rayleigh Fading model with a signal to noise ratio of 60 dB. For 50 dB, the results for SchedEx are reported to be on average better than those for the two-phase enhancer. SchedEx produces the schedules in runtimes 2-3 orders of magnitude faster than the two-phase approach. The simulations in [12] are conducted for topologies of sizes between 50 and 200 uniformly at random distributed sensors, for single-path routing over a single channel, given a single sink in the center of a circular area.

As explained in [12], slotted scheduling algorithms operate as follows.

(i) **While** not all packet buffers are empty,

- (1) **apply scheduling algorithm** to decide the next slot(s),
- (2) **append the slot(s)** to the schedule frame,
- (3) **update packet buffers** according to the transitions,
- (4) update meta-data (if required).

SchedEx substitutes step 3 for packet update to ensure reliability by a controlled packet move delay. The authors

in [12] therefore propose Algorithm 1 which maintains a vector of down-counters for all transceivers in the network, based on $\vec{\tau}$. Given $\vec{\tau}$, an initial attempt vector is created where each transmitter contains the amount of attempts per packet to ensure a link reliability which is calculated given the user-defined end-to-end reliability demand $\underline{\rho}$. Each time a transmission attempt for a sensor has been registered in a slot, its counter is decremented. SchedEx only registers a packet move from a transmitter to a receiver, once its down-counter reaches 0, which even triggers a reinitialization of the counter thereafter to the required amount of attempts per transmission. The algorithm terminates once all packets have arrived at their destinations, identical to the none reliability-aware scheduling version.

The lower-bound formulation of minimal repetitions n_t per transceiver from [12] to ensure the end-to-end reliability $\underline{\rho}$ is here repeated for convenience. It applies to networks utilizing a single-path routing table R and assesses a total amount of $k_1, \dots, k_{|\mathcal{T}|}$ packets passing each transceiver t , $1 \leq t \leq \mathcal{T}$ on their way towards any of the sinks via multihop. An end-to-end reliability of $\underline{\rho}$ is guaranteed if n_t repetitions are executed for each route from t to *parent* R_t for each packet over the whole network, according to

$$n_t = \left\lceil \frac{\log(1 - \underline{\rho}^{1/(|\mathcal{T}| \cdot k_t)})}{\log(1 - q_{tR_t})} \right\rceil, \quad \forall t \in \mathcal{T}. \quad (11)$$

Given the lower bound n_t for a single packet, a repetition vector $\vec{\tau}$ can be created with $\tau_t = n_t$. If each packet transmission through t ensures at least τ_t attempts, the end-to-end reliability of the network, ρ , will be larger or equal to the demand $\underline{\rho}$.

6. Method

SchedEx calculates the minimal amount n_t (11) of repetition of attempts to ensure a minimum end-to-end reliability $\underline{\rho}$ over the entire WSN, given the node-to-node packet reception rates between all sensors. n_t is then integrated algorithmically into the scheduling algorithm to ensure $\underline{\rho}$ for superframes produced by arbitrary scheduling algorithms. The improvements made in this paper are outlined below.

6.1. Multichannel Assignment. Each centralized slot-based scheduling algorithm, in Step 1 of the loop, sequentially adds transceivers to the slot until a conflict, according to constraints c_4 or c_5 , emerges. We extend the routine that checks the conflicts for single channel networks by a loop that iterates all channels for a new candidate transmitter and only report a conflict if the transmitter conflicts for all available channels in $\{1, \dots, CH_{MAX}\}$. Otherwise, the transceiver is scheduled for the first nonconflicting channel, in numeric order.

6.2. SchedEx Improvement. In this section, we introduce a new formula that substitutes (11), prove its correctness, and show that the adjusted formula requires the proven lowest amount of repetitions per scheduled link for SchedEx and any given initial buffer b .

Proposition 1. *Given a network with a single-path routing table R , sinks \mathcal{S} , transceivers \mathcal{T} , a link quality matrix Q , and a total amount of $k_1, \dots, k_{|\mathcal{T}|}$ packets passing each transceiver t , $1 \leq t \leq \mathcal{T}$ on their way towards any of the sinks via multihop. An end-to-end reliability of $\underline{\rho}$ can be guaranteed if n_t repetitions are executed for each route from t to R_t for each packet over the whole network, according to*

$$n_t = \left\lceil \frac{\log(1 - \underline{\rho}_t)}{\log(1 - q_{tR_t})} \right\rceil, \quad \forall t \in \mathcal{T} \quad (12)$$

with

$$\underline{\rho}_t = \underline{\rho}^{1/\hat{k}}, \quad (13)$$

$$\hat{k} = \sum_{t \in \mathcal{T}} k_t. \quad (14)$$

Proof. For the case of one link and k packets, this is trivially true because of Proposition 2 in [12]. For $|\mathcal{T}|$ links, and k_t transmissions for each $t \in \mathcal{T}$, we get, according to (14), a total of \hat{k} stochastically independent transmissions in the network. We therefore need to guarantee

$$\underline{\rho} \leq \prod_{i=1}^{\hat{k}} \underline{\rho}_i, \quad (15)$$

to make (12) fulfilled (3) for each transmission. However, this is fulfilled because of

$$\prod_{t \in \mathcal{T}} \underline{\rho}_t = \prod_{i=1}^{\hat{k}} \underline{\rho}_i = \underline{\rho}^{\hat{k}} = (\underline{\rho}^{1/\hat{k}})^{\hat{k}} = \underline{\rho}. \quad (16)$$

Due to the equality of $\prod_{t \in \mathcal{T}} \underline{\rho}_t$ and $\underline{\rho}$, we get that (13) is tightest bound for SchedEx producing a minimal demand of repetitions n_t per transceiver $t \in \mathcal{T}$, guaranteeing end-to-end reliability constraint $\underline{\rho}$. \square

7. Simulations

We apply single-path routing in this paper. Routing tables are determined by the Dijkstra shortest path algorithm using the expected transmission count (ETX) metric [27]. Table 1 contains the network model parameters.

TABLE 1: Simulation setup parameters.

Parameter	Value
Routing	Single-path
Routing algorithm	Shortest path (ETX)
Network size	{50, 100, 200, 400, 800}
Channel model	Rayleigh Fading
a_n	67.7328
g_n	0.9819
γ_{pn}	4.2935
α	3.3
λ	0.5
SNR (γ_n)	60 dB
Transmission range r_t	30 units
Interference range r_i	60 units

7.1. Topologies. A simulation model with n randomly distributed nodes in a circular area of radius 100 units around a center point at (100, 100) is applied. The first $n\lambda/(\lambda + 1)$ nodes are distributed uniformly at random within the inner circle of radius $100/\sqrt{2}$, with $\lambda \in [0, 1]$. The remaining nodes are randomly distributed in the outer partition of the circle. All sensors in \mathcal{T} have a packet waiting for transmission; thus, $b = \vec{1}$.

We analyze 50 topologies, each ten of sizes 50, 100, 200, 400, and 800. SchedEx supports WSNs with multiple sinks, while the simulations only use a single sink. We extend the work by assessing each topology in three variations, with

- (1) a single sink in the center at (100, 100),
- (2) two sinks, one at (50, 50) and one at (150, 150),
- (3) four sinks at {(50, 50), (50, 150), (150, 50), (150, 150)}.

For the multisink topologies, all sinks are connected via a backbone-network, which is why the choice of sink for transmission does not matter for the sensors. We investigate all scenarios first without reliability constraint and then for reliability demand $\underline{\rho} \in \{0.9, 0.999, 0.99999\}$. All topologies with their calculated link quality levels are publicly available (<https://github.com/feldob/wsnScenarios/tree/master/SchedEx%2B>). We utilize four scheduling algorithms, namely, node-based scheduling [8], level-based scheduling [8], dedicated scheduling [13], and shared scheduling [13], following the same setup. The interference range is modeled by directed packet reception rates of 0.00001, assuring the avoidance of transmissions for interfering sensors.

8. Results

8.1. Multiple Channels, Reliability-Unaware. Figure 1 illustrates the impact of the amount of utilized channels on latency and runtime. In terms of schedule size the performance is similar for all algorithms. Node-based scheduling provides the majority of shortest schedules for the given topologies, with a runtime that is between 50 and 220% faster than for the competitors. For one and two channels, node-based scheduling is clearly better, whereas the variance in the amount

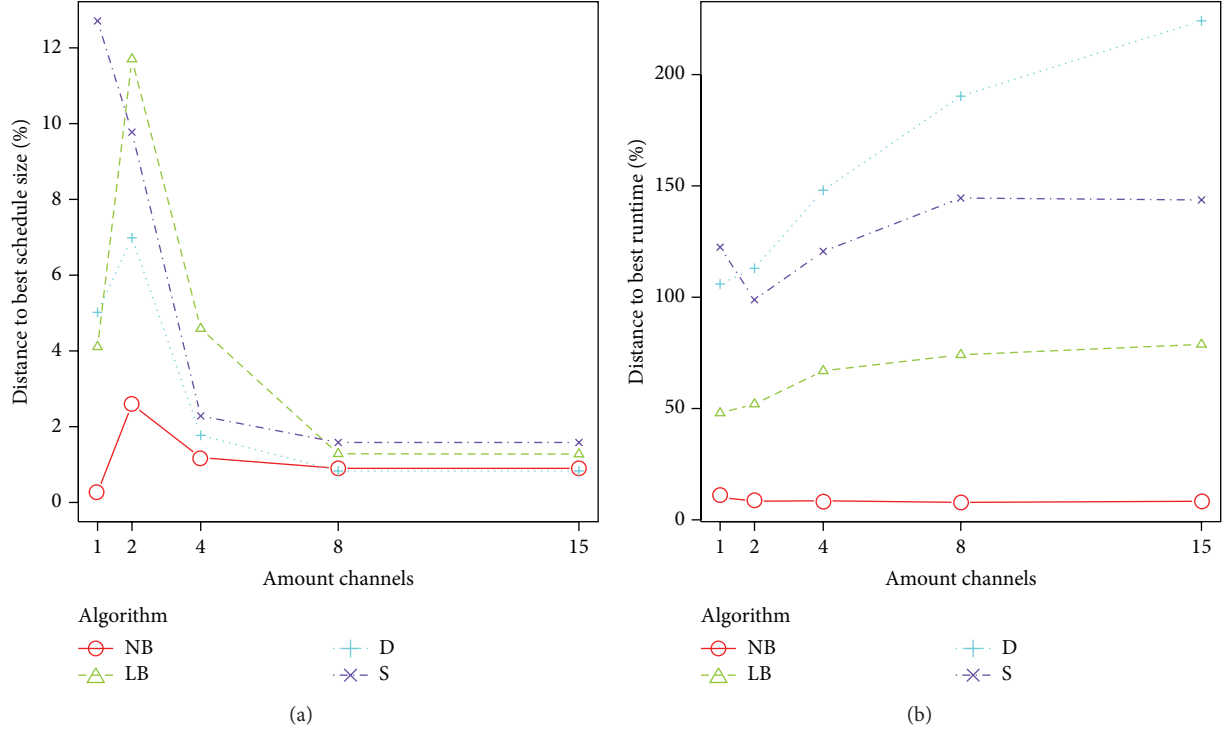


FIGURE 1: The average impact of the amount of available channels on the scheduling performance (a) and runtime (b) of the benchmarked scheduling algorithms. The more channels are utilized, the less impact has the choice of scheduling algorithm on the latency performance.

of utilized channels. For 8 and 15 channels the expected performance is very similar, also because the results converge towards the theoretical optimum $\lceil |\mathcal{T}|/|\mathcal{S}| \rceil$. The dedicated scheduling algorithm scales worst in the amount of channels with respect to runtime.

8.2. Multiple Channels, Reliability-Aware. We define the simulations using n_t from (11) as SchedEx(1) and n_t from (12) as SchedEx(2). Table 2 presents the reliability-aware scheduling results for the two SchedEx variants and the two-phase approach proposed in [13]. Figure 2 presents the scheduling and runtime performance for the competitors in comparison. With few exceptions, SchedEx(2) is best in terms of minimizing latency, while guaranteeing reliability, inspite of the amount of channels. SchedEx(1) performance slightly decreases in the amount of channels utilized, with schedules that are about 20% longer than the best ones identified (almost exceptionally by SchedEx(2)). The schedules from the two-phase approach are throughout about 10–16% longer than the best ones found. SchedEx(1) has longer execution times compared to SchedEx(2) by in average ca. 20%. The schedules are in average 20% longer which explains the increase in runtime, since both are based on the same algorithm. The two-phase approach is on average 35–55 times slower than SchedEx(2).

8.3. Multiple Sinks. The impact for the one, two, and four sink scenarios is presented in Figure 3(a) for basic scheduling and Figure 3(b) for reliability-aware scheduling. The results

show that all algorithms react similar to the introduction of multiple sinks. For two sinks, schedule size can on average be reduced by ca. 40% and for four sinks by ca. 67%. For SchedEx(1), schedule size reduces relatively better in the amount of sinks, compared to the competitors. SchedEx(2) reduces latency by 47% for two instead of one sink and by 67% using four sinks. SchedEx(2) produces on average the shortest schedules for all scenario types, even though the competitors are catching up the higher the number of utilized sinks.

8.4. Runtime Impact. Figures 4(a) and 4(b) illustrate the impact of multiple sinks and multiple channels, respectively, on the runtime of the scheduling. Where the amount of sinks has a large impact on the scheduling performance, the runtime is less strongly affected. The amount of applied channels, however, has a much larger impact on the algorithm runtime than the amount of sinks. The runtime increases substantially until a certain point. Thereafter, a performance decline can be observed.

9. Discussion

We are interested in how the introduced formula (12) compares to the original one and how the scheduling algorithms scale in the amount of sinks and utilized channels. From the nonreliability-aware simulations, we derive that single-channel simulations, as conducted in previous studies, are not sufficient to evaluate the scheduling efficiency of an algorithm for real-life scenarios where multiple channels are available.

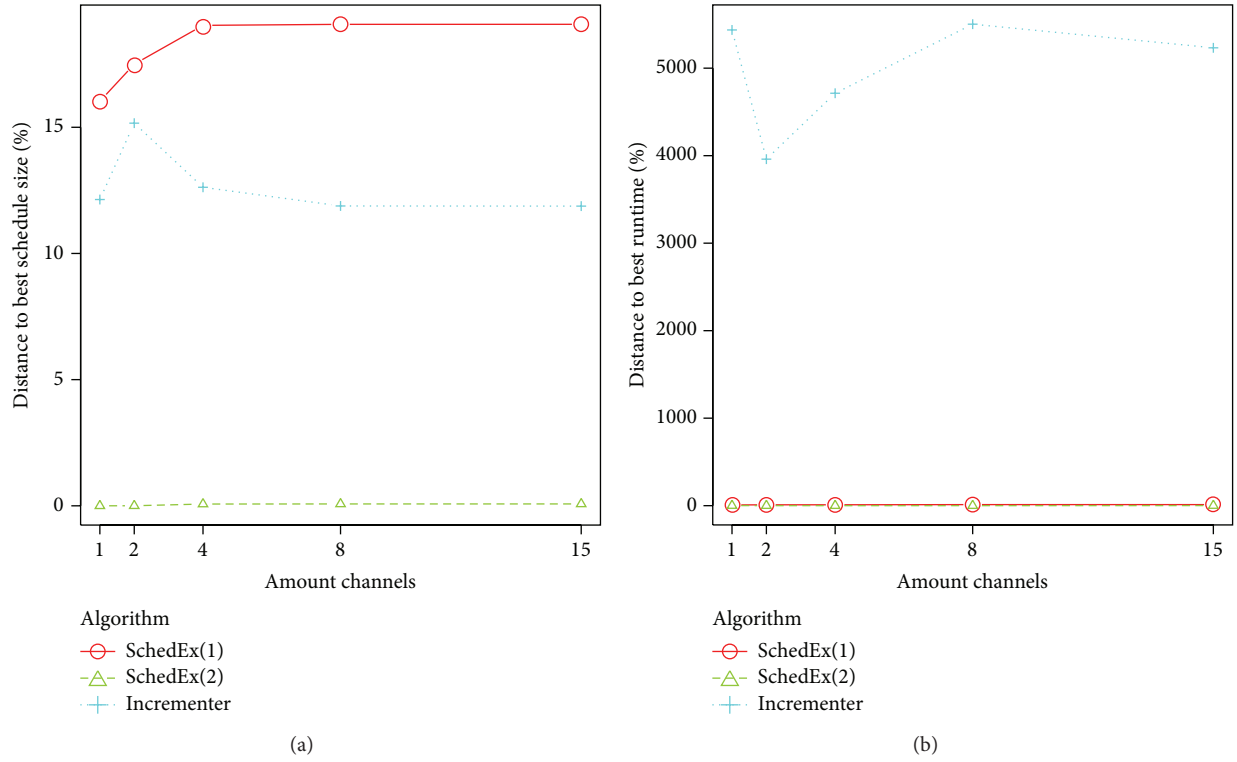


FIGURE 2: The average impact of the amount of available channels on the scheduling performance (a) and runtime (b) for the three reliability-aware approaches. SchedEx(1) produces the shortest schedules throughout the investigation.

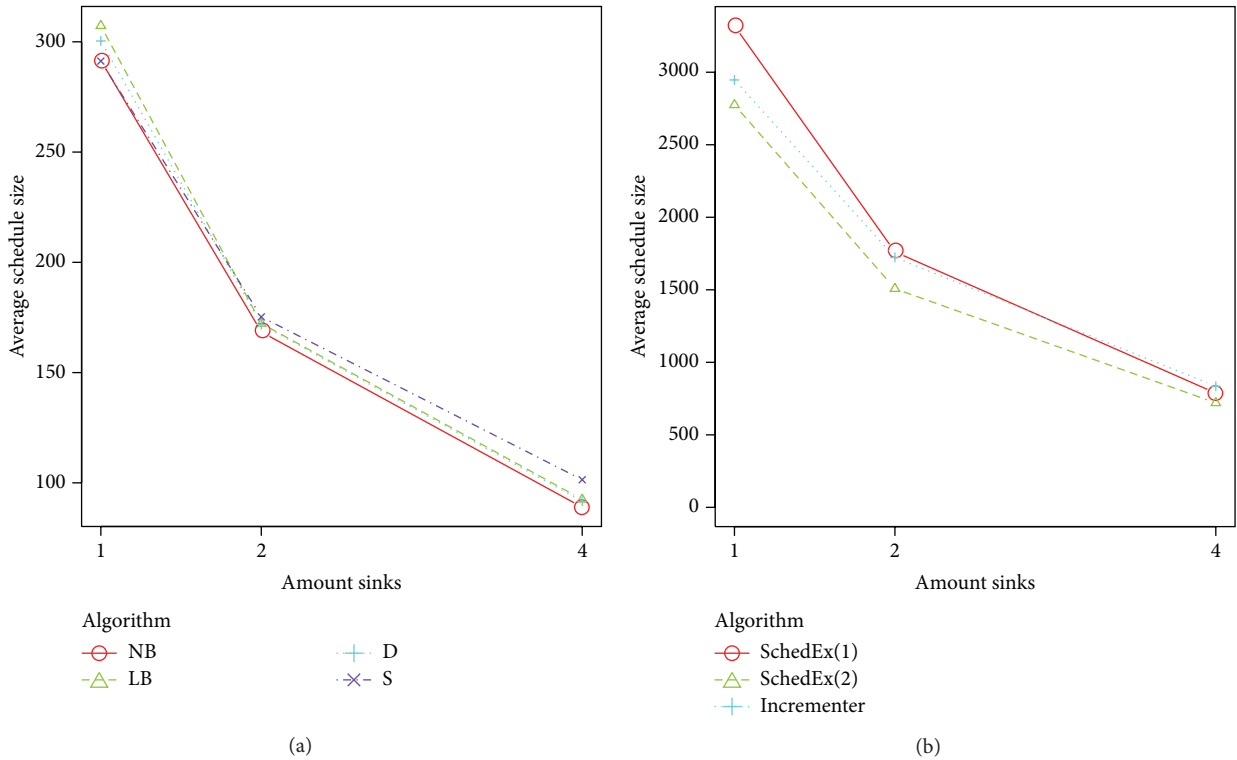


FIGURE 3: The average impact of the amount of available sinks on latency for the four scheduling algorithms (a) and the three reliability-aware scheduling approaches (b). Reliability-aware scheduling performance is equally positively correlated with the amount of deployed sinks in the network compared to basic scheduling.

TABLE 2: Results for node-based scheduling given the networks of 5 different sizes, 1–15 utilized channels, and a stringent reliability requirement of $\rho = 0.99999$.

Reliability algorithm		SchedEx(1)		SchedEx(2)		Two-phase approach	
Size	Channels	ms	$ \mathcal{F} $	ms	$ \mathcal{F} $	ms	$ \mathcal{F} $
1 sink topology							
50	1	7 ± 1	1423 ± 469	7 ± 1	1223 ± 368	863 ± 512	1364 ± 416
	2	5 ± 1	869 ± 267	5 ± 1	745 ± 215	472 ± 244	877 ± 257
	4	4 ± 1	761 ± 229	4 ± 1	648 ± 181	393 ± 191	755 ± 208
	8	5 ± 1	759 ± 225	4 ± 1	646 ± 181	389 ± 192	755 ± 208
	15	6 ± 1	759 ± 225	5 ± 1	646 ± 181	390 ± 191	755 ± 208
200	1	85 ± 4	3648 ± 112	79 ± 2	3232 ± 98	6779 ± 501	3512 ± 95
	2	62 ± 6	2565 ± 140	59 ± 10	2241 ± 119	3760 ± 239	2440 ± 102
	4	44 ± 4	2307 ± 182	39 ± 1	2003 ± 155	3164 ± 282	2148 ± 147
	8	38 ± 3	2286 ± 186	34 ± 1	1985 ± 159	3054 ± 266	2115 ± 147
	15	40 ± 3	2286 ± 186	36 ± 1	1985 ± 159	3063 ± 258	2115 ± 147
800	1	1644 ± 36	14925 ± 204	1605 ± 68	13228 ± 176	353375 ± 36084	14226 ± 147
	2	1102 ± 17	10653 ± 238	1052 ± 9	9309 ± 197	135011 ± 8600	10051 ± 181
	4	704 ± 12	9879 ± 200	670 ± 12	8569 ± 172	105797 ± 7189	9008 ± 141
	8	505 ± 8	9856 ± 199	477 ± 5	8546 ± 176	106114 ± 9437	8888 ± 148
	15	514 ± 8	9856 ± 199	483 ± 7	8546 ± 176	105777 ± 8073	8886 ± 147
4 sink topologies							
50	1	4 ± 1	366 ± 85	4 ± 1	343 ± 78	83 ± 24	392 ± 71
	2	3 ± 0	229 ± 74	2 ± 0	214 ± 67	52 ± 16	257 ± 58
	4	2 ± 0	198 ± 63	2 ± 0	183 ± 57	45 ± 16	211 ± 52
	8	2 ± 0	198 ± 63	2 ± 0	182 ± 56	42 ± 13	209 ± 50
	15	2 ± 0	198 ± 63	2 ± 0	182 ± 56	43 ± 13	209 ± 50
200	1	49 ± 1	1037 ± 74	48 ± 2	978 ± 63	926 ± 89	1153 ± 72
	2	34 ± 2	604 ± 41	33 ± 2	563 ± 35	533 ± 51	706 ± 42
	4	24 ± 4	536 ± 30	21 ± 1	500 ± 25	453 ± 29	604 ± 21
	8	20 ± 2	534 ± 29	19 ± 1	498 ± 25	445 ± 23	596 ± 18
	15	22 ± 3	534 ± 29	20 ± 1	498 ± 25	447 ± 29	596 ± 18
800	1	999 ± 25	4147 ± 141	984 ± 21	3921 ± 120	19189 ± 1211	4519 ± 130
	2	651 ± 20	2409 ± 102	632 ± 18	2256 ± 78	9434 ± 453	2761 ± 92
	4	367 ± 9	2184 ± 94	359 ± 11	2027 ± 82	8079 ± 309	2436 ± 65
	8	281 ± 5	2181 ± 93	271 ± 7	2023 ± 83	8014 ± 323	2413 ± 67
	15	283 ± 7	2181 ± 93	277 ± 10	2023 ± 83	8022 ± 297	2414 ± 66

The performance for the four algorithms is very different for one to four channels, and node-based scheduling seems to offer a good trade-off between frame-size and execution time. The overall general schedule size in Figure 3(a) suggests that the choice of a best scheduling algorithm with respect to schedule frame size is hard, because the summarized statistics reveal that all competitors perform on average very similar. For the reliability-aware simulations, the new formula (12) significantly outperforms the one from [12]. It is substantially faster in execution time and obtains better schedules than both SchedEx(1) and the two-phase approach from [13], with narrow variances. For the one-channel one-sink utilization on topologies of size 50 and reliability constraint $\rho = 0.99999$ SchedEx(1) in [12] requires an expected runtime of 14.3 seconds, assuming WirelessHART with slot-sizes of 10 ms. SchedEx(2) utilizing four sinks and four channels in the same

setup produces a 20% shorter schedule within 2 ms and reduces the expected runtime to 1.83 seconds, which implies a substantial improvement of nearly one order of magnitude. Assuming the same setup as for [12], applying formula (11), SchedEx performs on average 14% better in terms of schedule frame-size, with a standard deviation which is about 21% more narrow.

10. Conclusions

The simulations presented in this paper reveal that the introduced SchedEx enhancements and problem formulation extensions lead to a substantial performance improvement. Significant improvements in terms of performance and run-time can be achieved using multiple sinks and utilizing multiple channels for reliable packet delivery in WSN. Still,

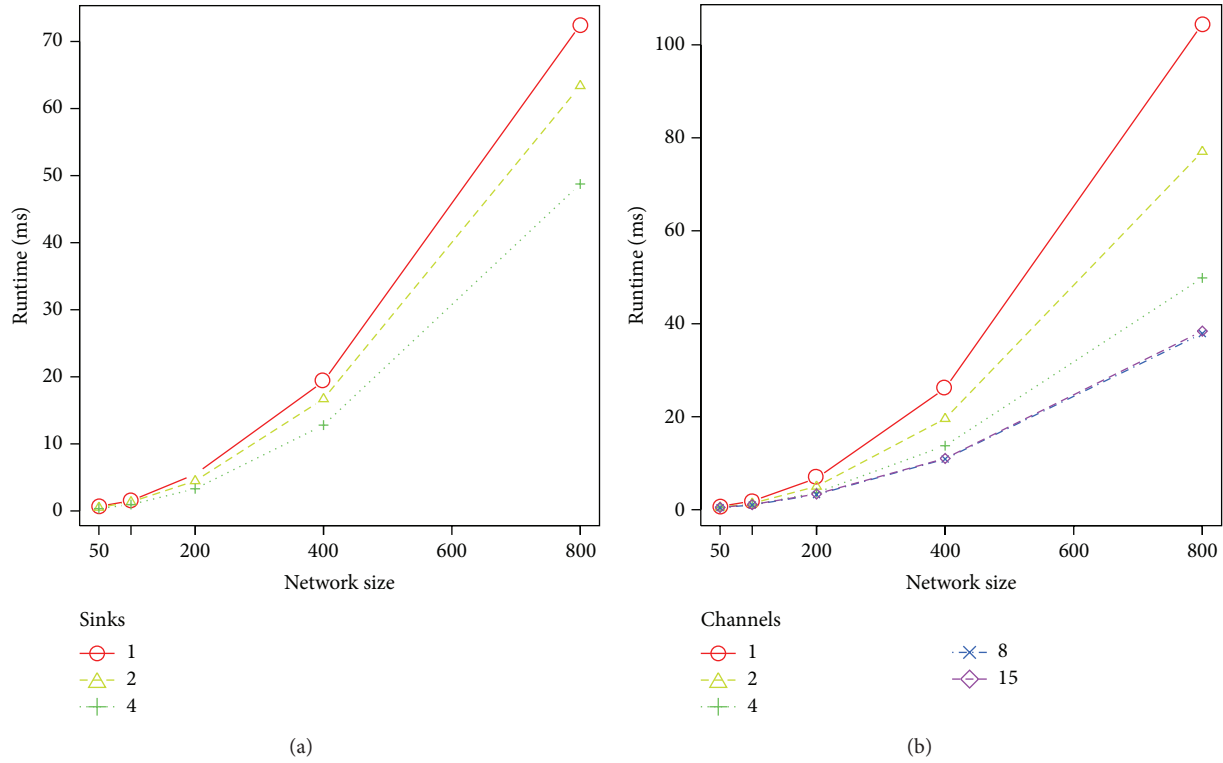


FIGURE 4: The average execution time impact of the amount of available sinks (a) and channels (b), respectively, on the scheduling algorithms. The choice of utilized scheduling algorithm becomes less important with respect to latency performance the more channels are available. Node-based scheduling however performs best in both execution time and latency.

there is space for improvement, and a need for future work. Adaptive algorithms that for a given topology are able to recommend a setup that includes the amount of channels to be used and also the number of sinks and their positioning for a simplified deployment are required. Our simulations suggest that the utilization of four channels is a good heuristic value for IWSN, in order to reach a close-to-optimal scheduling performance and as fast a runtime as possible. In real scenarios, the distribution of sensors is usually not uniform, and positioning of sinks might be difficult due to application specific constraints. Where objective functions for network design can be formulated, heuristic algorithms can identify near-optimal solutions that fulfill the given constraints.

The identification of the routing table R could become part of the problem formulation in order to better address the network dynamics. Weighted shortest path routing may be far from being the most appropriate routing table for a topology given specific transmission patterns and interference landscapes.

Further, a WSN specific risk analysis for reliable packet delivery is required which for the given deployment calculates the risk of failure over a time period. Then, for real networks, flows are usually arbitrary and maximum acceptable latencies are on a sensor, if not packet, level. SchedEx in its extended version significantly improves on the applicability of reliability-aware scheduling for IWSN, where link qualities in the network can vary in time, and new schedules have to quickly be computed and deployed.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] V. Ç. Güngör and G. P. Hancke, *Industrial Wireless Sensor Networks: Applications, Protocols, and Standards*, CRC Press, New York, NY, USA, 2013.
- [2] P. Suriyachai, U. Roedig, and A. Scott, "A survey of MAC protocols for mission-critical applications in wireless sensor networks," *IEEE Communications Surveys and Tutorials*, vol. 14, no. 2, pp. 240–264, 2012.
- [3] A. A. K. Somappa, K. Øvsthus, and L. M. Kristensen, "An industrial perspective on wireless sensor networks—a survey of requirements, protocols, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1391–1412, 2014.
- [4] S. Ergen and P. Varaiya, "Tdma scheduling algorithms for sensor networks," Tech. Rep., Berkeley University, Berkeley, Calif, USA, 2005.
- [5] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.
- [6] T. Lennvall, S. Svensson, and F. Hekland, "A comparison of WirelessHART and ZigBee for industrial applications," in *Proceedings of the 7th IEEE International Workshop on Factory Communication Systems (WFCS '08)*, pp. 85–88, May 2008.

- [7] "WirelessHART specification 75: Tdma data-link layer, rev. 1.1," Tech. Rep. hCF_SPEC-75, HART Communication Foundation Standard, 2008.
- [8] S. C. Ergen and P. Varaiya, "TDMA scheduling algorithms for wireless sensor networks," *Wireless Networks*, vol. 16, no. 4, pp. 985–997, 2010.
- [9] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "Real-time scheduling for wirelessHART networks," in *Proceedings of the IEEE 31st Real-Time Systems Symposium (RTSS '10)*, pp. 150–159, IEEE, San Diego, Calif, USA, November–December 2010.
- [10] Ö. B. Akan and I. F. Akyildiz, "Event-to-sink reliable transport in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 5, pp. 1003–1016, 2005.
- [11] A. Liu, D. Zhang, P. Zhang, G. Cui, and Z. Chen, "On mitigating hotspots to maximize network lifetime in multi-hop wireless sensor network with guaranteed transport delay and reliability," *Peer-to-Peer Networking and Applications*, vol. 7, no. 3, pp. 255–273, 2014.
- [12] F. Dobslaw, M. Gidlund, and T. Zhang, "End-to-end reliability-aware scheduling for wireless sensor networks," *IEEE Transactions on Industrial Informatics*. In press.
- [13] M. Yan, K.-Y. Lam, S. Han et al., "Hypergraph-based data link layer scheduling for reliable packet delivery in wireless sensing and control networks with end-to-end delay constraints," *Information Sciences*, vol. 278, pp. 34–55, 2014.
- [14] S.-E. Yoo, P. K. Chong, D. Kim et al., "Guaranteeing real-time services for industrial wireless sensor networks with IEEE 802.15.4," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, pp. 3868–3876, 2010.
- [15] W. Shen, T. Zhang, F. Barac, and M. Gidlund, "PriorityMAC: a priority-enhanced MAC protocol for critical traffic in industrial wireless sensor and actuator networks," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 824–835, 2014.
- [16] N. Marchenko, T. Andre, G. Brandner, W. Masood, and C. Bettstetter, "An experimental study of selective cooperative relaying in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 3, pp. 1806–1816, 2014.
- [17] J. Chen, Q. Yu, B. Chai, Y. Sun, Y. Fan, and X. S. Shen, "Dynamic channel assignment for wireless sensor networks: a regret matching based approach," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 1, pp. 95–106, 2015.
- [18] S. He, J. Chen, P. Cheng, Y. Gu, T. He, and Y. Sun, "Maintaining quality of sensing with actors in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pp. 1657–1667, 2012.
- [19] S. He, J. Chen, D. K. Y. Yau, and Y. Sun, "Cross-layer optimization of correlated data gathering in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 11, pp. 1678–1691, 2012.
- [20] P. Cheng, S. He, F. Jiang, Y. Gu, and J. Chen, "Optimal scheduling for quality of monitoring in wireless rechargeable sensor networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 3072–3084, 2013.
- [21] P. Djukic and S. Valaee, "Delay aware link scheduling for multi-hop TDMA wireless networks," *IEEE/ACM Transactions on Networking*, vol. 17, no. 3, pp. 870–883, 2009.
- [22] W. Shen, T. Zhang, M. Gidlund, and F. Dobslaw, "SAS-TDMA: a source aware scheduling algorithm for real-time communication in industrial wireless sensor networks," *Wireless Networks*, vol. 19, no. 6, pp. 1155–1170, 2013.
- [23] G. Fiore, V. Ercoli, A. J. Isaksson, K. Landernäs, and M. D. di Benedetto, "Multihop multi-channel scheduling for wireless control in wirelessHART networks," in *Proceedings of the IEEE Conference on Emerging Technologies and Factory Automation (ETFA '09)*, pp. 1–8, September 2009.
- [24] J. Zhao, Y. Qin, D. Yang, and Y. Rao, "A source aware scheduling algorithm for time-optimal convergecast," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 251218, 12 pages, 2014.
- [25] P. Suriyachai, J. Brown, and U. Roedig, "Time-critical data delivery in wireless sensor networks," in *Proceedings of the 6th IEEE International Conference on Distributed Computing in Sensor Systems*, pp. 216–229, 2010.
- [26] S. Munir, S. Lin, E. Hoque, S. M. S. Nirjon, J. A. Stankovic, and K. Whitehouse, "Addressing burstiness for reliable communication and latency bound generation in wireless sensor networks," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN '10)*, pp. 303–314, April 2010.
- [27] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.

