Impacts of Channel Selection on Industrial Wireless Sensor-Actuator Networks

Dolvara Gunatilaka†, Mo Sha‡, Chenyang Lu†
†Cyber-Physical Systems Laboratory, Washington University in St. Louis
‡Department of Computer Science, State University of New York at Binghamton

***Abstract*—Industrial automation has emerged as an important application of wireless sensor-actuator networks (WSANs). To meet stringent reliability requirements of industrial applications, industrial standards such as WirelessHART adopt Time Slotted Channel Hopping (TSCH) as its MAC protocol. Since every link hops through all the channels used in TSCH, a straightforward policy to ensure reliability is to retain a link in the network topology only if it is reliable in all channels used. However, this policy has surprising side effects. While using more channels may enhance reliability due to channel diversity, more channels may also reduce the number of links and route diversity in the network topology. We empirically analyze the impact of channel selection on network topology, routing, and scheduling on a 52-node WSAN testbed. We observe inherent tradeoff between channel diversity and route diversity in channel selection, where using an excessive number of channels may negatively impact routing and scheduling. We propose novel channel and link selection strategies to improve route diversity and network schedulability. Experimental results on two different testbeds show that our algorithms can drastically improve routing and scheduling of industrial WSANs.**

I. INTRODUCTION

Wireless sensor-actuator networks (WSANs) have been increasingly adopted as a communication infrastructure for process industries such as oil refineries and chemical plants. Wired networks are often costly to deploy and maintain in industrial environments, and they are also difficult to reconfigure to accommodate new production process requirements. In contrast, WSANs make it easy to retrofit existing industrial facilities without running cabling for communication and power. IEEE 802.15.4 based WSANs are designed to operate at low data rates and can be manufactured inexpensively, which makes them ideal for industrial process applications running on battery-powered embedded systems where energy consumption and costs are often important. Industrial applications have critical demands for reliable and real-time communication in harsh environments, which pose distinctive challenges for WSANs in preserving the stability and control performance of a plant. Failing to meet these demands can lead to production inefficiency, safety threats, and financial loss. These stringent requirements differentiate industrial WSANs from traditional wireless sensor networks (WSNs), which require only best effort service [1].

With almost a decade of real-world deployments, the WirelessHART standard [2] has shown it can achieve reliable low-power wireless communication in industrial environments and has become a leading wireless standard for industrial process applications. To support reliable and real-time communication in harsh environments, WirelessHART incorporates a set of specific design choices. For instance, WirelessHART adopts a centralized network architecture. It employs a TSCH (Time Slotted Channel Hopping) [3] at the MAC layer on top of the IEEE 802.15.4 physical layer. It also supports channel blacklisting to avoid poor channels, which is the focus of this paper.

WirelessHART uses a network-wide channel blacklisting approach in which all devices in the network blacklist the same set of channels. The network operator performs by measuring the wireless channel's condition and then manually blacklisting channels where a large number of external wireless signals were observed during the measurement. Unfortunately, this manual approach is labor intensive and error prone.

TSCH uses a channel hopping mechanism in which each link in the network hops through all channels that are not blacklisted, i.e., the channels used. To ensure reliability, a straightforward approach is to use only links that are highly reliable in all channels used for transmissions. Hence, channel selection can significantly impact the network topology. As a result, while using more channels can enhance reliability due to channel diversity and can increase the number of transmissions that may potentially be scheduled in a time slot, more channels may also lower route diversity in the network topology. Channel selection must therefore balance channel diversity and route diversity in a WirelessHART network.

To our knowledge, this paper represents the first systematic study on channel selection for WirelessHART networks. We believe our findings and designs are applicable to other industrial WSAN standards based on TSCH networks. The contributions of this work are three-fold:

- We empirically study the impact of channel selection on network topology, routing, and real-time performance based on the topology of a WSAN testbed. We find the performance of a WirelessHART network does not improve monotonically with an increasing number of channels used, due to the tradeoff between channel and route diversity.
- Based on the insights gathered from our studies, we propose a channel selection algorithm and a link selection strategy which can configure the network at deployment and/or runtime in response to operating dynamics. Our algorithms take the guesswork out of the channel

blacklisting process through an automated selection of channels to balance channel diversity and route diversity.
- We evaluate our channel selection algorithms using a wireless testbed running a WirelessHART protocol stack. Experimental results show our channel selection algorithms can significantly improve the network's capability to meet the routing and scheduling demands of data flows, while requiring only moderate processing time.

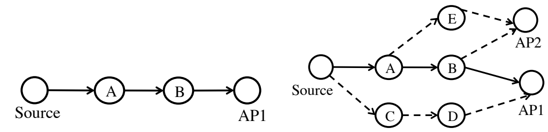
The remainder of the paper is organized as follows. Section II introduces the background of WirelessHART networks, and Section III presents our empirical studies. Sections IV and V describe our channel selection algorithm and link selection strategy. Section VI presents our evaluation. Section VII reviews related work, and Section VIII concludes the paper.

II. Background of WirelessHART Networks

A WirelessHART network consists of a gateway, multiple access points, and a set of field devices (sensors and actuators) forming a multi-hop mesh network. The access points and field devices are equipped with radio transceivers compatible with the IEEE 802.15.4 physical layer. The access points are wired to the gateway and serve as bridges between the gateway and the wireless network. A WirelessHART network is managed by a centralized network manager, a software module residing on the gateway responsible for generating routes and schedules and maintaining the network. The centralized architecture enhances the visibility and predictability of network operations.

TSCH MAC: Time Slotted Channel Hopping technology has been implemented as a MAC protocol, and was introduced as part of the IEEE 802.15.4e standard in 2012 for the industrial process control and automation. WirelessHART employs the TSCH MAC that offers deterministic and collision-free communication. All nodes in a TSCH network are time synchronized, and time is divided into 10 ms time slots, which are sufficient to accommodate a packet transmission and its acknowledgment (ACK). A time slot can either be *dedicated* or *shared*. In a dedicated slot, only one transmission is allowed in each channel while in a shared slot, two senders compete for the channel for transmitting to a common receiver in a CSMA/CA fashion. TSCH operates on the 2.4 GHz ISM band and can use up to 16 channels defined in the IEEE 802.15.4 standard. To mitigate the impact of interference and enhance channel diversity, TSCH adopts a channel hopping mechanism where each device switches to a new channel in every slot. While our work focuses on WirelessHART networks, our findings on the impacts of channels on route diversity are applicable to other TSCH-based networks.

Channel Blacklisting: To avoid using channels with poor quality, WirelessHART adopts a *network-wide* channel blacklisting approach that allows the network operator to manually blacklist channels with observed poor performance. This network-wide blacklisting policy ensures all channels used for communication are highly reliable and that all links used for transmissions use this same set of channels.



(a) Source Routing. (b) Graph Routing.

Fig. 1. Source and graph routing (solid lines represent primary paths and dashed lines represent backup paths).

Network Protocol: WirelessHART supports two routing approaches, named *source routing* and *graph routing*. Source routing provides a single route for each data flow from a source to a destination node. Graph routing consists of a single primary path and a backup path for each node on a primary path. As shown in Figure 1, for source routing, to send a packet to an access point AP1, a source node must transmit a packet through nodes A and B. Under graph routing, the packet may take backup routes to reach the access points if links on a primary path fail to deliver a packet. Compared to source routing, graph routing offers a higher degree of reliability through route diversity and transmission redundancy.

III. Empirical Study

In this section, we introduce our experimental setup and present our empirical studies on the impact of channel selection on network topology, routing, and scheduling.

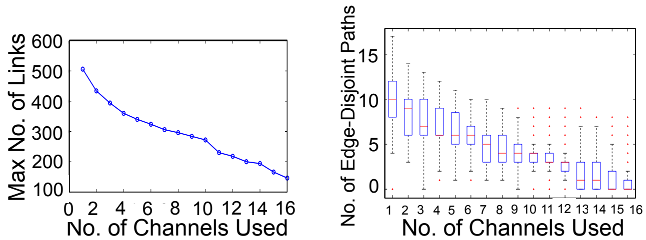
A. Empirical Study Settings

To investigate the impacts of channel selection on WSAN performance, we collect topology information on all 16 IEEE 802.15.4 channels from the *WUSTL* testbed [4] at Washington University in St. Louis, consisting of 52 TelosB motes spanning across two adjacent buildings. In our study, two nodes are designated as access points and the rest serve as field devices.

Channel Selection: We use the packet reception ratio (PRR) to estimate link reliability. PRR is defined as the fraction of transmitted packets successfully received by the receiver. Following a straightforward link selection policy, the WSAN only use bidirectional links whose PRRs are above a threshold in *all* the channels used and in both directions. This is because each link cycles through all the channels used through channel hopping. As a result, a link with a poor PRR in any of the channels used can incur packet loss at run time. Additionally, the link needs to be bidirectional, supporting reliable communication of both data packets and their acknowledgments.

To assess the impacts of channel selection on the performance of a WirelessHART network, we use a brute-force search algorithm to identify the set of channels that offers the maximum number of available links. Given k channels used, we compute all possible combinations of k channels from the 16 IEEE 802.15.4 channels and choose the set of channels that allows the most links to remain in the network topology. We set the PRR threshold PRR_t to 80% in this study.

Routing and Scheduling: Consider a set of real-time periodic flows sharing a WirelessHART network, each of which delivers a packet from a source to a destination. The network



(a) Maximum number of available links.
(b) Number of edge-disjoint paths.

Fig. 2. Impact of the number of channels used on the network topology.

manager finds the shortest path as the primary path of a flow for both source and graph routing. For backup paths of graph routing, we run the same shortest path algorithm to obtain the backup route from each node on a primary path to the destination. The network manager then schedules transmissions of each flow based on the rate monotonic policy [5], a fixed priority scheduling often adopted by industrial WSANs. Under the rate monotonic policy, a flow with a shorter period has a higher priority. The period of each flow is selected within the range $2^{0\sim7}$ seconds, which falls within the common range of periods used in process industries, and its deadline equals its period.

B. Impact on Routing

In this subsection, we first study the impact of the number of channels used on the network topology and routing. As Figure 2(a) shows, the number of available links *decreases* with an increasing number of channels used. More channels used result in fewer available links due to the stringent reliability requirement, in which the PRRs of these links must be no less than the desired threshold over all channels used. We further quantify the route diversity of a network by computing the number of edge-disjoint paths of a flow, i.e., the number of paths from a source node to a destination node that does not share any common edge. We generate 580 flows. Figure 2(b) presents box plots of the number of edge-disjoint paths per flow. Increasing the number of channels used degrades the route diversity in the resulting topology.

We then investigate how the number of channels used affects routing. We stipulate that a source route is successfully generated for a flow set if at least one primary route exists for each flow in this flow set. A graph route is successfully generated if at least one primary route exists from the source to the destination of each flow in the flow set and at least one backup route exists from each node along the primary route to the destination. We generate 100 sets of flows by varying the locations of sources and destinations. If a route is successfully generated for all flows in a flow set, then route generation deems successful. The success rate of route generation is defined as the fraction of flow sets with successful routes.

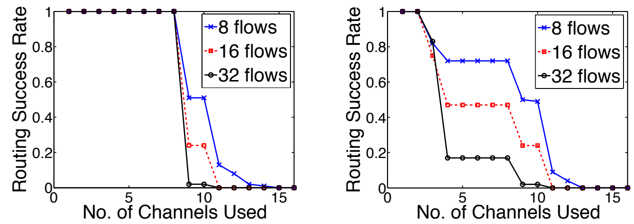
Figure 3(a) and Figure 3(b) present the success rates of route generation under source and graph routing, respectively. The routing success rate decreases with an increasing number of channels used due to its impact on network topology. The number of channels used has a particularly large impact on graph routing, given its dependency on route diversity. We repeat our studies on another testbed (the Indriya testbed [6]) and with another routing algorithm (the energy-efficient routing algorithm [7]) and observe similar results and the same impact of the number of channels used on the network topology and routing. We omit these results for brevity and refer to an extended report [8].

Observation 1*: More channels used reduce the route diversity of a network, resulting in a lower success rate of route generation. Note that more channels used have the benefit of enhancing channel diversity. Our findings show that channel selection must strike a balance between channel and route diversity.*

C. Impact on Scheduling

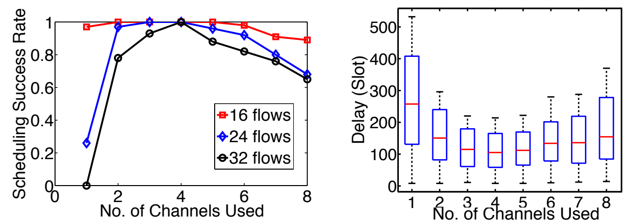
In this section, we study how the number of channels used affects transmission scheduling. Once the routes of a flow set are successfully generated, the network manager computes a transmission schedule by assigning time slots and channel offsets to all transmissions of the flows. The transmission schedule generated by the network manager is called a *superframe* and is disseminated to field devices. At run time, field devices repeat the schedule when they reach the end of the superframe.

A flow is *schedulable* if all its packets will meet its deadline, and a set of flows is schedulable if all the flows are schedulable. Flows are scheduled based on priorities assigned by the rate monotonic algorithm. The scheduling success rate is defined as the fraction of flow sets that are schedulable among flow sets with successful route generation.



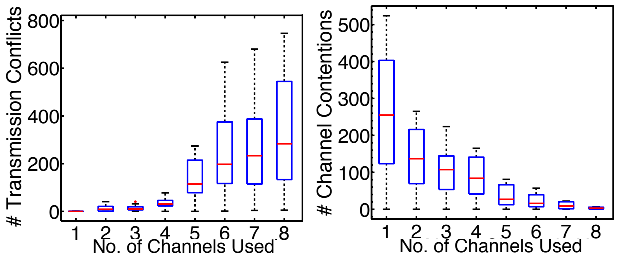
(a) Source Routing.
(b) Graph Routing.

Fig. 3. Route genration success rates, based on the shortest path algorithm.



(a) Scheduling success rate under graph routing
(b) Box plots of worst-case delays of 32 flows under graph routing.

Fig. 4. Schedule generation success rates and worst-case delays.



(a) Transmission Conflict. (b) Channel Contention.

Fig. 5. Number of transmission conflicts and channel contentions of the 32-flow schedule under graph routing.

As Figure 4(a) shows, with 32 flows, the scheduling success rate increases when the number of channels used increases from 1 to 4. This is to be expected, because more channels allow more concurrent transmissions. Surprisingly, the trend reverses when the scheduling success rate *decreases* with more channels used. We observe the same trend in the worst-case packet delays of 32 flows, as shown in Figure 4(b). The minimum delay occurs when the number of channels is 4. (When the number of channels exceeds 8 in our experiments, the routing success rate with 32 flows is very small. Thus, we do not have sufficient sets of flows to determine the scheduling success rate with more than 8 channels used.)

We find that this phenomenon results from the tradeoff between two sources of delays in a WirelessHART network [1]: channel contention and transmission conflict:

1) *Channel contention*: To avoid interference, WirelessHART allows only one transmission per channel in each time slot. If the number of transmissions scheduled in a time slot exceeds the number of channels, lower-priority transmissions must be delayed to later slots.
2) *Transmission conflict*: Due to the half-duplex nature of IEEE 802.15.4 radios, a node can participate in only one transmission or one reception in the same slot. Consequently, transmissions conflict if they involve a common node as either a sender or a receiver. Then, only one of the conflicting transmissions can be scheduled in that slot.

Figure 5(a) and Figure 5(b) present the number of transmission conflicts and the number of channel contentions in our previous experiment. Interestingly, while channel contention decreases with more channels used, transmission conflicts increase as more transmissions share nodes, due to reduced route diversity. Our results also suggest that transmission conflict dominates when the number of channels used is beyond a certain point, so adding more channels can be detrimental.

Observation 2*: Because both transmission conflict and channel contention affect scheduling, schedulability does not improve monotonically with more channels used; in fact, using more channels may degrade schedulability.*

IV. Channel Selection

In Section III, a brute-force approach is used to search for the set of channels that retains the maximum number of available links in the network topology. However, this approach is impractical due to its high computational cost, especially because the network manager needs to adjust the selected channels at run time due to changes in wireless conditions. Moreover, maximizing the total number of links does not always lead to maximum route diversity because different nodes may not contribute equally to route diversity. For example, a node adjacent to fewer links may be more vulnerable to channel selection. If its few links are eliminated due to channel selection, the node may be removed from the network topology. To overcome these challenges, we introduce a new channel selection algorithm, which operates in three steps: (1) removing channels on which critical nodes are poorly connected to the network (**Channel Filtering**), (2) ranking the channels (**Channel Ranking**), and (3) finding the maximum number of channels that can be used to successfully generate the routes and schedule for a given set of flows. In contrast to the brute-force approach, our channel selection algorithm employs a new metric for ranking channels that considers the criticalities of nodes to the network topology.

We model a WSAN as a set of graphs $G = \{G_i(V, E_i) \mid 11 \le i \le 26\}$, where i is the channel index [1]. Each node $v \in V$ denotes a device, and each edge $e_i \in E_i$ denotes a bidirectional link with PRRs no lower than a given PRR threshold in both directions on channel i. We define the degree of v on channel i, $D_i(v)$, as the number of edges incident to v on channel i.

Channel Filtering: We first perform channel filtering to ensure that the channel selection retains all critical nodes, including the sources and destinations of flows and the access points connected to the gateway. A channel i is filtered out if there exists a critical node v whose node degree $D_i(v) < 3$ on this channel, as WirelessHART suggests that any device in a network should have at least three neighbors for route diversity [2].

Channel Ranking (CR): For each node, a channel is preferable if the node has a higher degree under the channel. CR normalizes the degree of node v on channel i by dividing $D_i(v)$ by the maximum degree of v among all channels to prevent discrimination against nodes with fewer links. The normalized degree of v is denoted $D_i'(v)$. Furthermore, if a node has fewer good channels, it is more likely to be eliminated from the network after channel selection. Channel i is considered a *good* channel for the node v if the following two conditions are met:

- $D_i(v) > \overline{D_i(v)}$, the degree of node v is larger than the average degree of all nodes on the channel i.
- $D_i(v) > 3$, the node v has more than 3 neighbors.

The number of good channels of a node v is denoted by $n_c(v)$. To avoid eliminating nodes with few good channels in channel selection, we include $n_c(v)$ in the channel score as follows:

[1] IEEE 802.15.4 standard specifies 16 channels (channel 11 $\sim$ 26) in the 2.4 GHz ISM band [9].

$$score(i) = \sum_v D'_i(v)/n_c(v) \quad (1)$$

The output of CR is an ordered list of channels $c_1, c_2, ...c_n$ sorted in decreasing order of $score(i)$. A channel with a higher rank potentially enhances route diversity compared to a channel with a lower rank.

Finding the Maximum Number of Channels Used: Because channel hopping enhances reliability through channel diversity, our channel selection approach searches for the *maximum* number of channels that allows routes to be successfully generated and all flows to meet their deadlines. Given a ranked list of n channels generated by the CR, we select the top k (initialized to $k = n$) channels from the ranked list, and then run routing and scheduling algorithms. If the route and schedule generation fail, the algorithm decreases k by 1 and reruns the routing and scheduling algorithms until it finds k that can successfully generate routes and a schedule for the flow set.

V. Link Selection

To meet the reliability requirement of Industrial WSANs, we blacklist links whose PRR is below a threshold in any of the channels used. The choice of the threshold involves a tradeoff between network reliability and route diversity. A higher threshold may improve reliability at the cost of route diversity. Figure 6 shows the impact of different PRR thresholds on the success rate of route generation. In comparison to a 90% PRR threshold, an 80% PRR threshold allows both source and graph routes to achieve a higher route generation success rate. Although a higher PRR threshold leads to a lower success rate, employing a lower PRR threshold may degrade the overall network reliability.

In contrast to traditional link selection based on a *single* PRR threshold, we propose a novel *Channel Pairing (CP)* strategy to maintain reliability while improving route diversity. CP takes advantage of the redundant transmissions offered by WirelessHART, which dictate that both the source and the graph routing perform a second transmission (retransmission) if the first attempt (transmission) fails when delivering a packet. Therefore, instead of using a single PRR threshold to ensure that selected links are reliable across all channels used, CP divides the channels used into two sets and applies two different PRR thresholds to these two sets of channels during link selection. A high PRR threshold is applied to the first set of channels that are dedicated to the first transmissions, while a lower PRR threshold is applied to the other set of channels used for retransmissions. Using more reliable channels for transmissions can reduce the chance of retransmissions, while allowing retransmissions to use less reliable channels helps preserve links and route diversity. CP ensures overall reliability by taking both transmissions into account.

A. Channel Pairing Algorithm (CP)

The input of CP is a ranked list of k channels obtained from the channel selection algorithm, C_r, and $P_{success}$, which is the

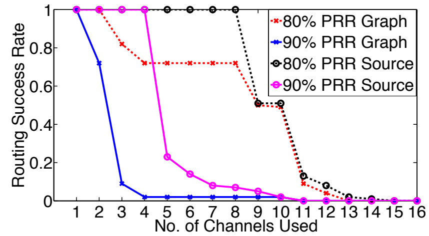


Fig. 6. Success rates of source and graph route generation with an 80% and a 90% PRR threshold.

required probability of successfully sending a packet after a transmission and a retransmission. In addition, CP allows two PRR thresholds, $PRR1_t$ and $PRR2_t$, for link selection on channels used for transmissions and retransmission, respectively, where $PRR1_t > PRR2_t$.

The outputs of CP are two sets of channels, $C_1 = \{c_{1,1}, c_{1,2}...c_{1,x}\}$ and $C_2 = \{c_{2,1}, c_{2,2}...c_{2,x}\}$, where $|C_1| = |C_2| = x$ and $x = \left\lfloor \frac{k}{2} \right\rfloor$. (If the number of channels used, k, is odd, CP defines c_{back}, which can be used by backup transmissions of graph routing. c_{back} is the middle channels in C_r. Every link in the output set of links L must also have a PRR no lower than $PRR1_t$ on c_{back}.) CP schedules the first transmission on more reliable channels in C_1 and retransmissions on channels in C_2. CP also outputs a set of links, L, selected for use in communication. For every link $l \in L$, if its first transmission attempt is on channel $c_{1,i}$, then its retransmission must happen on channel $c_{2,i}$. The PRR of a link l on a channel c is defined as $PRR(l_c)$. Every link $l \in L$ must satisfy the following *link quality constraints*:

1) $PRR(l_{c_{1,i}}) \geq PRR1_t$, $\forall c_{1,i} \in C1$
2) $PRR(l_{c_{2,i}}) \geq PRR2_t$, $\forall c_{2,i} \in C2$
3) $1 - (1 - PRR(l_{c_{1,i}}))(1 - PRR(l_{c_{2,i}})) \geq P_{success}$ $\forall c_{1,i} \in C1$ and $c_{2,i} \in C2$ (The two transmissions are independent due to channel hopping.)

Constraint (3) allows a communication over a link to meet the reliability requirement ($P_{success}$) through the transmission and its retransmissions. Constraint (1) ensures that the links have a high PRR in the first transmission and hence reduces the likelihood of the retransmission and saves energy consumption by communication devices. Constraint (2) allows the retransmission to utilize a lower quality link when allowed by constraint (3).

CP works in three steps. It first splits the channels in C_r into two sets. $C1$ contains the top x channels in C_r, and C_2 has the remaining channels. CP considers a set of links L_1, in which each link $l \in L_1$ meets constraint (1). It then computes an average PRR of all links in $L1$ for each channel $c_{1,i} \in C1$ and sorts the channels in $C1$ in increasing order of the average PRRs.

In the second step, for each channel $c_{1,i}$ in a sorted $C1$, CP pairs it with a channel $c_{2,i}$ in $C2$. The selected $c_{2,i}$ pairing with $c_{1,i}$ must maximize the number of links meeting all three link quality constraints. $c_{2,i}$ must also be at least h hops away from $c_{1,i}$. If no such channel exists, $c_{2,i}$ is the channel with the maximum hopping distance from $c_{1,i}$. Hopping to a nearby channel may not be effective in improving reliability due to the significant correlation between transmission failures among adjacent channels [10].

Lastly, let the set of valid links of a pair of channels $c_{1,i}$ and $c_{2,i}$ be $L_{(c_{1,i},c_{2,i})}$. Then, the final output is a set of links $L_{(c_{1,1},c_{2,1})} \cap L_{(c_{1,2},c_{2,2})}, ... \cap L_{(c_{1,x},c_{2,x})}$, where every selected link meets the link quality constraint in every channel pair.

B. Channel Scheduling for CP

In this section, we describe how to adjust a scheduler to support channel pairing. In WirelessHART networks, the network manager generates one or more superframes, where each superframe comprises a transmission schedule that is repeated in a cyclic fashion. To avoid interference, each time slot can accommodate at most k concurrent transmissions, where k is the number of channels used. At run-time, each sender and receiver pair switches to the same channel to communicate. The standard calculates the channel hopping sequence based on the following formula:

$$LogicalChannel = (ASN + choffset)\%k \qquad (2)$$

where ASN (Absolute Slot Number) denotes the cumulated slot number since the network starts and k is the number of channels used. The channel offset $choffset$ is assigned by the scheduler during transmission schedule computation ($0 \leq choffset \leq k-1$). It guarantees that no transmissions in the same slot use the same channel. The sender and receiver then map a logical channel on to a physical channel using a common mapping table stored in a node.

In a straightforward link selection policy, where a link must be equally reliable on all channels used, a transmission can use any of these channels. However, CP needs to guarantee that, given two sets of channels C_1 and C_2, if the first transmission attempt over a link uses $c_{1,i} \in C_1$, then its retransmission must use $c_{2,i} \in C_2$. Therefore, we impose a channel assignment constraint when scheduling a transmission. For transmissions on a primary path, if the first transmission is assigned to a slot number s_1 and channel offset $choffset_1$, then, for its retransmission, the scheduler needs to find a slot s_2 and a channel offset $choffset_2$ such that

$$(s_1 + choffset_1)\%x = (s_2 + choffset_2)\%x \qquad (3)$$

A channel offset in each slot is in the range [0, $x-1$], where $x = |C_1| = |C_2|$. The constraint ensures that a transmission and its retransmission always use the logical channel pair $c_{1,i}$ and $c_{2,i}$, where $0 \leq i \leq x-1$. In addition, for the backup transmissions of graph routing, they can use c_{back} or any available channel from C_1 and C_2 since CP relies on the two transmission attempts on a primary path to achieve the desired reliability. c_{back} is preferable to a channel from C_1 and C_2 because it is not used by transmissions on the primary path.

VI. Evaluation

We evaluate our approach in four aspects: (1) routing, (2) scheduling, (3) execution time, and (4) network reliability. To demonstrate the generality of our approach, we evaluate the routing performance using three real network topologies of different sizes and locations: (1) Full Testbed - a 52-node WUSTL testbed deployed in two connected buildings; (2) Half Testbed - part of the WUSTL testbed including only 32 nodes in one of the buildings; (3) Indriya - an open-access testbed deployed on three floors at the National University of Singapore [6], consisting of 86 nodes. We compare our solutions CR alone and CR+CP against two baseline approaches: (1) ML, searching for the set of channels used that maximizes the number of available links by computing all channel combinations; (2) ML(Rank), ranking channels based on the number of available links and then selecting the k highest ranked channels as the channels to be used.

A. Success Rate of Route Generation

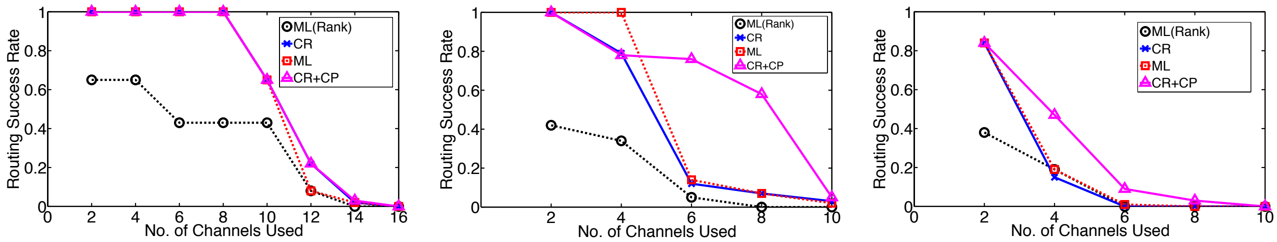
To evaluate the effectiveness of our algorithms in enhancing route generation, we run experiments with 100 flow sets, each of which contains eight flows that have different sources and destinations. We generate routes based on the shortest path algorithm. Because an industrial WSAN usually employs reliable links, we set the PRR threshold to 90% for ML, ML(Rank), and CR. Given a transmission and a retransmission using a uniform PRR threshold of 90%, the overall reliability of a link is 99%. We hence set the $P_{success}$ = 99% for CP to achieve the same level of reliability. $PRR1_t$ and $PRR2_t$ are set to 90% and 70%, respectively.

Figure 7(b) and Figure 8(b) show the route generation success rates under source and graph routing on the Full Testbed. A more flexible link selection strategy allows CR+CP to significantly outperform ML(Rank), CR, and ML, especially under graph routing, which relies heavily on route diversity. Figure 7(a) and Figure 8(a) show similar results in which CR+CP again outperforms the two baselines on the smaller but better connected Half Testbed. CR+CP on both testbeds significantly improves the route generation success rate when the number of channels used is high (between 6 - 10), thereby allowing the network to maintain more channel diversity.

As Figure 7(c) and Figures 8(c) show, on the Indriya testbed, CR+CP likewise achieves a higher success rate of route generation than the two baselines for both source and graph routings. CR+CP offers less improvement in the route generation success rate on the Indriya testbed than on the other two testbeds due to weaker network connectivity. It can also be observed that CR outperforms ML(Rank) on all testbeds for both source and graph routings, due to a more effective ranking criteria. CR also achieves a route generation success rate comparable to ML, but with less complexity. Although CR adopts a better heuristic than ML, CR still has a drawback when compared to ML that explores all combinations of channels.

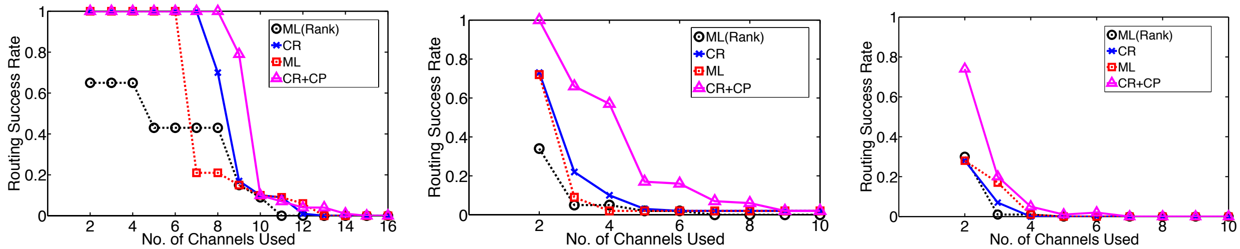
B. Algorithm Execution Time

We measure the execution time of our algorithms on a MacBook Pro laptop with a 2.7 GHz Intel Core i7. The channel selection process includes finding the best set of



(a) Half Testbed (b) Full Testbed (c) Indriya Testbed

Fig. 7. Success rates of route generation under source routing.



(a) Half Testbed (b) Full Testbed (c) Indriya Testbed

Fig. 8. Success rates of route generation under graph routing.

TABLE I
CHANNEL SELECTION ALGORITHM EXECUTION TIME (IN MS).

Approach	Half Testbed	Full Testbed	Indriya
ML(Rank)	26.6	59.6	190.0
CR	55.7	121.4	324.9
ML	1297.2	2133.8	4676.5
CR+CP	83.5	156.5	420.5

TABLE II
SEARCH AND SCHEDULE GENERATION TIME (IN MS).

Approach	8 flows	16 flows	32 flows
ML	38.47	63.75	125.8
CR+CP	42.07	66.18	120.2

channels for each k ($1 \leq k \leq 16$) channels used, and obtaining the output topology containing links that satisfy the reliability requirement. Table I presents the execution time (in milliseconds) of ML(Rank), CR, ML, and CR+CP on the Half Testbed, the Full Testbed, and the Indriya testbed. (Because the WirelessHART standard [2] suggests the network with a single gateway should comprise at most 80 field devices, the Indriya testbed represents a WirelessHART network with the maximum size.) CR incurs an approximately 2X higher execution time than ML(Rank) in all testbeds due to a more sophisticated ranking criterion. CR+CP consumes up to 3.1X and 1.5X longer execution time than ML(Rank) and CR because it introduces additional computations for pairing channels. ML finds the best set of channels through a brute-force search on all channel combinations, hence it incurs 15.5X, 13.6X, and 11.1X longer execution times than CR+CP on the Half Testbed, the Full Testbed, and the Indriya testbed, respectively.

We then find the maximum number of channels that can be used to generate a schedule. The number of flows significantly affects the execution time of this process. Table II compares the search time (in milliseconds) consumed by CR+CP and ML on the Indriya testbed. (We limit the number of flows to 32 due to the network size. Almost 80% of the nodes are designated as either a sender or a receiver.) As the number of flows increases, the execution time also grows under both CR+CP and ML. With the same number of flows, CR+CP and ML require similar amounts of execution time. CR+CP is more efficient than ML since it requires less overall execution time, but it achieves a higher route generation success rate.

C. Network Performance

Because CR+CP improves the success rate of route generation by allowing links to use less reliable channels for packet retransmissions, it may degrade the performance of the network. In this set of experiments, we compare the performance of CR+CP against ML, which employs a single PRR link selection ensuring all links are reliable across all channels used. We perform experiments on the Full Testbed to evaluate our approaches in term of the network reliability, the total number of transmissions of each schedule, and the number of transmission attempts per hop. The testbed runs the implementation of key network features of the WirelessHART standard, including TSCH MAC and a routing layer that supports both source and graph routings on TinyOS 2.1.2 and TelosB motes [4]. For each channel selection approach, we run experiments under graph routing with three different network settings. Each setting contains eight distinct flows and two access points. Each experiment is run for 100 rounds using eight IEEE 802.15.4 channels. Both ML and CR+CP follow formula (2) in Section V-B when computing a channel hopping sequence.

We use the end-to-end packet delivery ratio (PDR) as the metric to quantify network reliability. As Table III shows,

TABLE III
NETWORK PERFORMANCE COMPARISON UNDER GRAPH ROUTING

Set	Approach	PDR	#Trans / hop	#Trans	#Hops / flow
1	ML(B)	99.25%	1.01	49.56	6.13
	CR+CP	99.88%	1.01	49.43	6.00
2	ML(B)	99.75%	1.02	70.82	8.63
	CR+CP	98.75%	1.05	57.61	6.88
3	ML(B)	99.00%	1.03	44.13	5.38
	CR+CP	98.38%	1.08	42.37	4.88

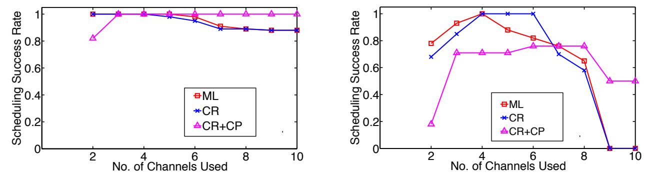
CR+CP achieves an average PDR lower than that of ML by approximately 1% in all settings, except under setting 1, where CR+CP surpasses ML by 1%. The slight decrease in the average PDR of CR+CP is in exchange for a proportionally much-higher success rate of route generation.

To determine the number of transmissions incurred by CR+CP, we measure the average number of transmission per hop. Graph routing allows two transmission attempts over a link on a primary path, and an additional transmission on a backup link. Table III shows that in setting 1, CR+CP requires the same average number of transmissions per hop as ML. Nevertheless, CR+CP performs slightly worse than ML, as observed in setting 2 and 3, where CR+CP has 3% and 5% increases in the average number of transmissions per hop compared to ML.

We further compute the average total number of transmissions needed for all flows to deliver packets from their sources to destinations, as shown in Table III. As ML requires fewer transmissions per hop than CR+CP, it should require fewer transmissions per schedule. However, the results demonstrate that CR+CP achieves a smaller total number of transmissions in all settings than ML, especially in setting 2, where CR+CP needs 22.9% fewer transmissions than ML. This is due to the fact that CR+CP enhances route diversity and increases the chance of the routing algorithm obtaining a shorter flow path. For instance, in setting 2, CR+CP reduces the average number of hops per flow by 20.2% over ML. Although CR+CP incurs more transmission attempts per hop than ML because of the use of weaker channels for retransmissions, it requires fewer transmissions per schedule than ML and maintains high reliability.

D. Success Rate of Schedule Generation

Because using more channels may have a negative effect on the scheduling success rate as the number of channels reaches a certain point, we investigate whether a similar trend can be observed under CR and CR+CP. We compare the scheduling success rate of our approaches against ML on the Full Testbed topology. In this set of experiments, we lower the PRR thresholds of ML and CR to 80%. For CP, $P_{success}$ is set to 96% and $PRR1_t$ and $PRR2_t$ are 80% and 70%, respectively. (To evaluate the success rate of schedule generation, the network needs to contain a large number of flows. However, increasing the number of flows under a high PRR threshold, e.g., 90%, significantly decreases the route generation success rate. Hence, there are not enough sets of flows to determine the schedule generation success rate.) We run the experiments with 100 sets of flows.



(a) 16 flows (b) 32 flows.

Fig. 9. Schedule generation success rates under graph routing on Full Testbed

As shown in Figure 9(a) and Figure 9(b), CR obtains a scheduling success rate comparable to ML under both 16 and 32 flows. With 32 flows, CR+CP leads to a lower schedule generation success rate than ML when the number of channels is within [2,6]. While CR+CP potentially reduces the number of transmission conflicts through route diversity, it also introduces a channel scheduling constraint that causes transmissions of flows to be delayed and consequently miss their deadlines. In contrast, when the number of channels exceeds 6, CR+CP outperforms ML for both 16 and 32 flows because CR+CP can fully take advantage of route diversity since it has enough channels to use. Note that when the number of channels reaches 9, both CR and ML can no longer successfully generate routes; therefore, their scheduling success rates drop to 0. CR+CP improves the success rate of schedule generation when there are more channels used, hence it enhances channel diversity. The number of channels used also has a similar impact on the success rate of schedule generation under CR and CR+CP. CR and CR+CP achieve the maximum success rate of schedule generation when the number of channels is within [4,6] and [6,9] respectively. Judicious selection of the number of channels is important to meet the timing constraints of WSANs. By exploring different numbers of channels automatically, our approaches can configure the channels efficiently.

VII. RELATED WORKS

There has been significant research on industrial WSANs spanning transmission scheduling, routing algorithms and network protocols, assuming a given set of channels. Readers are referred to recent review articles for comprehensive surveys [1] [11]. However, the impact of channel selection has not been studied in the context of industrial WSANs. Our work on channel selection is therefore complementary to previous work in this area.

802.15.4 radio link quality has been studied extensively in the context of WSNs. Sha et al. [10] performed empirical studies on real-world 2.4 GHz ISM band usage and channel conditions in residential environments. Ortiz et al. [12] evaluated the multi-channel behavior of 802.15.4 networks in a machine room, and an office testbed, and found route diversity to be an effective strategy to ensure reliability. Neither of these studies analyzed the impact of the number and the choice of channels on the route diversity of the network.

Multi-channel communication and channel allocation have been explored extensively in the wireless sensor network literature. For instance, Kim et al. [13] developed a multi-channel TDMA MAC protocol, and Raman et al. [14] designed a TDMA-based approach that uses channel hopping to improve network throughput. Wu et al. [15] and Vedantham et al. [16] proposed channel assignment schemes that partition a network into subgraphs and then assign channels to each subgraph. Tang et al. [17] and Le et al. [18] developed a multi-channel MAC protocol with dynamic channel allocation. Doddavenkatappa et al. [19] developed a channel selection and switching strategy that transforms intermediate quality links into good ones. Mobashir et al. [20] proposed a channel selection strategy in which only 3 channels far apart are used for communication. Chowdhury et al. [21] designed a scheme for packet scheduling and channel selection utilizing a carrier sensing mechanism. There also exists research on channel assignment for multi-radio multi-channel communication in the context of the IEEE 802.11-based wireless mesh networks, such as [22]–[29]. These works focused on solving the problem of optimizing multi-channel communication assuming a given set of channels, while our work investigates complementary problem of how to select a good set of channels. In contrast to the existing work designed for best effort service, our solution is tailored for industrial applications that demand reliable and real-time communication over the TSCH-based networks.

VIII. Conclusions

This paper investigates the tradeoff between channel and route diversity, based on empirical studies on testbeds. Our findings reveal that while using more channels means more channel diversity, a large number of channels may reduce route diversity, with negative impacts on routing and scheduling. As a result, using more channels is not always desirable in industrial WSANs. Based on these insights, we present novel channel and link selection algorithms that automatically select channels to facilitate routing and scheduling. Experimental results show that our solution significantly improves the success rates of routing and scheduling, while maintaining network reliability.

Acknowledgment

This work is supported, in part, by the NSF through grants 1320921 (NeTS), 1646579 (CPS), and 1657275 (CRII).

References

[1] C. Lu, A. Saifullah, B. Li, M. Sha, H. Gonzalez, D. Gunatilaka, C. Wu, L. Nie, and Y. Chen, "Real-Time Wireless Sensor-Actuator Networks for Industrial Cyber-Physical Systems," *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1013–1024, 2016.

[2] WirelessHART, 2007. [Online]. Available: http://www.hartcomm2.org

[3] IEEE 802.15.4e Time-Slotted Channel Hopping (TSCH). [Online]. Available: https://tools.ietf.org/html/rfc7554

[4] M. Sha, D. Gunatilaka, C. Wu, and C. Lu, "Implementation and Experimentation of Industrial Wireless Sensor-Actuator Network Protocols," in *EWSN*, 2015.

[5] J. W. S. Liu, *Real-Time Systems (1st ed.)*. Prentice Hall PTR, 2000.

[6] M. Doddavenkatappa, M. C. Chan, and A. L. Ananda, "Indriya: A Low-Cost, 3D Wireless Sensor Network Testbed," in *TRIDENTCOM*, 2011, pp. 302–316.

[7] C. Wu, D. Gunatilaka, A. Saifullah, M. Sha, P. B. Tiwari, C. Lu, and Y. Chen, "Maximizing Network Lifetime of WirelessHART Networks under Graph Routing," in *IoTDI*, 2016, pp. 176–186.

[8] Impacts of Channel Selection on Industrial Wireless Sensor-Actuator Networks. [Online]. Available: https://www.dropbox.com/s/guhhyzwublrpj5d/whextended.pdf?dl=0

[9] *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)*, IEEE, 2006.

[10] M. Sha, G. Hackmann, and C. Lu, "Real-world Empirical Studies on Multi-Channel Reliability and Spectrum Usage for Home-Area Sensor Networks," *IEEE Transactions on Network and Service Management*, vol. 10, no. 1, pp. 56–59, 2013.

[11] T. Watteyne, V. Handziski, X. Vilajosana, S. Duquennoy, O. Hahm, E. Baccelli, and A. Wolisz, "Industrial Wireless IP-Based Cyber-Physical Systems," *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1025–1038, 2016.

[12] J. Ortiz and D. Culler, "Multichannel Reliability Assessment in Real World WSNs," in *IPSN*, 2010, pp. 162–173.

[13] Y. Kim, H. Shin, and H. Cha, "Y-MAC: An Energy-Efficient Multi-channel MAC Protocol for Dense Wireless Sensor Networks," in *IPSN*, 2008, pp. 53–63.

[14] B. Raman, K. Chebrolu, S. Bijwe, and V. Gabale, "PIP: A Connection-Oriented, Multi-Hop, Multi-Channel TDMA-based MAC for High Throughput Bulk Transfer," in *SenSys*, 2010, pp. 15–28.

[15] Y. Wu, J. A. Stankovic, T. He, and S. Lin, "Realistic and Efficient Multi-Channel Communications in Wireless Sensor Networks," in *Infocom*, 2008.

[16] R. Vedantham, S. Kakumanu, S. Lakshmanan, and R. Sivakumar, "Component Based Channel Assignment in Single Radio, Multi-Channel Ad Hoc Networks," in *MobiCom*, 2006, pp. 378–389.

[17] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "EM-MAC: A Dynamic Multichannel Energy-Efficient MAC Protocol for Wireless Sensor Networks," in *MobiHoc*, 2011, p. 23.

[18] H. K. Le, D. Henriksson, and T. Abdelzaher, "A Practical Multi-Channel Media Access Control Protocol for Wireless Sensor Networks," in *IPSN*, 2008, pp. 70–81.

[19] M. Doddavenkatappa, M. C. Chan, and B. Leong, "Improving Link Quality by Exploiting Channel Diversity in Wireless Sensor Networks," in *RTSS*, 2011, pp. 159–169.

[20] M. Mobashir, G. XiangFa, and C. Choon, "Oppcast: Exploiting Spatial and Channel Diversity for Robust Data Collection in Urban Environments," in *IPSN*, 2016, pp. 1–12.

[21] K. R. Chowdhury and I. F. Akyildiz, "Interferer Classification, Channel Selection and Transmission Adaptation for Wireless Sensor Networks," in *ICC*, 2009.

[22] M. Alicherry, R. Bhatia, and L. Li, "Joint Channel Assignment and Routing for Throughput Optimization in Multi-Radio Wireless Mesh Networks," in *MobiCom*, 2005, pp. 58–72.

[23] C. Hua, S. Wei, and R. Zheng, "Robust Channel Assignment for Link-Level Resource Provision in Multi-Radio Multi-Channel Wireless Networks," in *ICNP*, 2008, pp. 157–166.

[24] X. Lin and S. Rasool, "A Distributed Joint Channel-Assignment, Scheduling and Routing Algorithm for Multi-Channel Ad-hoc Wireless Networks," in *Infocom*, 2007, pp. 1118–1126.

[25] B. Raman, "Channel Allocation in 802.11-Based Mesh Networks," in *Infocom*, 2006, pp. 1–10.

[26] K. N. Ramachandran, E. M. Belding, K. C. Almeroth, and M. M. Buddhikot, "Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks," in *Infocom*, 2006, pp. 1–12.

[27] S. Pediaditaki, P. Arrieta, and M. K. Marina, "A Learning-Based Approach for Distributed Multi-Radio Channel Allocation in Wireless Mesh Networks," in *ICNP*, 2009, pp. 31–41.

[28] J. Tang, G. Xue, and W. Zhang, "Maximum Throughput and Fair Bandwidth Allocation in Multi-Channel Wireless Mesh Networks," in *Infocom*, 2006, pp. 1–10.

[29] K. Xing, X. Cheng, L. Ma, and Q. Liang, "Superimposed Code Based Channel Assignment in Multi-Radio Multi-Channel Wireless Mesh Networks," in *MobiCom*, 2007, pp. 15–26.