DeCoT: A Dependable Concurrent Transmission-Based Protocol for Wireless Sensor Network

XIAOYUAN MA1,2, (Student Member, IEEE), PEILIN ZHANG4, (Student Member, IEEE), XIN LI1,3, (Student Member, IEEE), WEISHENG TANG1,2, (Student Member, IEEE), JIANMING WEI1, (Member, IEEE), AND OLIVER THEEL4, (Member, IEEE)
1Shanghai Advanced Research Institute, Chinese Academy of Sciences, China (e-mail: {maxy, lixin01, tangws, wjm}@sari.ac.cn)
2University of Chinese Academy of Sciences, China
3School of Information Science & Technology, ShanghaiTech University, China
4Department of Computer Science, Carl von Ossietzky University of Oldenburg, Germany (e-mail: {peilin.zhang, theel}@informatik.uni-oldenburg.de)
Corresponding author: Jianming Wei (e-mail: wjm@sari.ac.cn).
This research was partly supported by National Key R&D Program of China (2016YFC0801500) and the German Research Foundation through the Research Training Group DFG-GRK 1765: System Correctness under Adverse Conditions (SCARE, www.uni-oldenburg.de/scare).

ABSTRACT Concurrent transmission (CT)-based wireless sensor networks, where nodes transmit at the same moment upon receiving successfully, begin to be applied to real-world scenarios. CT-based protocols have been proven experimentally that they can achieve good the end-to-end performance, namely high reliability, low latency, and high energy efficiency. For various communication patterns (one-to-many, many-to-one and many-to-many), most current CT-based networks require a given and fixed host to realize global synchronization and scheduling. However, in real-world cases, there is a great deal of interference in the 2.4 GHz ISM band. Interference can partition the network unexpectedly due to the centralized scheduling in current CT-based networks. Even worse, current CT-based networks cannot complete the initialization phase if the unexpected partition occurs at the very beginning. To address this problem, we propose a dependable CT-based protocol (DeCoT) for WSN to support information exchange under adverse conditions. In DeCoT, a continuous transmission with channel hopping mechanism maintains links under interference and an initiated mechanism decentralizes the network.

Through our experiments in FlockLab, under interference, DeCoT achieves an average reliability of 87%, and outperforms the state-of-the-art flooding protocol – Robust Flooding that won the 1st place in the EWSN 2017 Dependability Competition. Especially when the sources are placed sparsely, DeCoT speeds up the information exchange. Above all, DeCoT can complete the initialization and work properly even when the network partitions unexpectedly. DeCoT is evaluated as the most reliable protocol in the EWSN 2018 Dependability Competition with respect to resistance against interference. In a word, DeCoT can work dependably under interference.

INDEX TERMS Concurrent transmission, wireless sensor networks, channel hopping, multichannel, dependable communication.

I. INTRODUCTION

With the development of wireless sensor networks (WSNs) in the past decades, WSN applications have covered a large number of aspects from precise agriculture to smart building. Most of these applications adopt the classical medium access control (MAC) model, Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA). Under the model of CSMA/CA, if the packets from two senders are transmitted at the same moment, the receiver, where two packets collide, hears nothing valid. The sender needs to listen to the channel before transmitting. If the channel is clear, then the sender transmits. Otherwise, the sender waits for a random period of time (i.e., due to a backoff mechanism) and re-transmits. The backoff increases the latency of the network. If a Radio Duty Cycle (RDC) strategy like ContikiMAC [1] is applied to save energy, then the timeliness
degrades further.

Concurrent transmission (CT), a novel MAC model has been introduced by Ferrari et al. [2]. In a CT-based network, there are two important properties: 1) constructive interference and 2) the capture effect [3]. Constructive interference means that radio signals of identical packets transmitted concurrently are superimposed rather than collided. The capture effect means that the radio signal with the greater amplitude can be received successfully rather than collided by others in frequency modulation (FM) systems. Generally, it is probably that the receiver receives one valid packet when packets are transmitted concurrently due to the two properties. This means that no more backoffs are needed and the latency is improved. CT-based networks also can achieve high reliability because no-logic topology is required. After that, more and more researchers started to apply CT-based WSN to some industrial scenarios requiring low latency, such as monitoring some simple events with a timely feedback control. To achieve more diverse communication patterns (i.e., many-to-one, one-to-many and many-to-many) simultaneously, state-of-the-art CT-based protocols like LWB [4], Chaos [5] and Crystal [6] generally rely on one host to initiate and schedule the whole network. These works have experimentally demonstrated that CT-based WSNs can improve the end-to-end performance, i.e., enhance reliability, lower latency and power consumption through an accurate global scheduling.

In order to let such promising networks work in a real application setting rather than in the laboratory, the complicated radio-frequency (RF) environment is one of the greatest challenges which ones are confronted with. Many existing and widely-used wireless communication devices (i.e., WiFi and Bluetooth) work in the 2.4 GHz ISM Band. Moreover, some electromagnetic radiations in this band are also introduced by some appliances like microwave ovens. All of these devices disturb the wireless channels of IEEE 802.15.4 [7] and affect the performance of practical systems. From the view of the network, under interference, the network could be partitioned unexpectedly into several isolated parts which cannot communicate with each other at all. This is unfavorable for the network which is scheduled by a fixed host because the global scheduling and synchronization packets could not reach all the partitions. Therefore, current CT-based WSNs are not able to work properly under strong interference due to unexpected network partitioning.

In summary, there are two reasons why a node cannot receive the scheduling and synchronization packet from the host in a current CT-based network: 1) the receiver is interfered by interference, or 2) the network is partitioned. When a receiver is interfered by interference, the low signal-to-noise ratio (SNR) blocks the receiver’s detection of the packet. In a partitioned network, all the paths between the host and the receiver are broken despite that it is clear round the receiver. For example, as shown in Figure 1, the gray nodes, which are far from the host H, are isolated and not able to receive packets from the host over time. Some nodes would lose synchronization if they are in an isolated part for a long time. What is worse, they can never be synchronized with the host if the network is partitioned during the network initialization phase. These unsynchronized nodes, which cannot relay packets, bring about degradations of the connectivity and reliability and consume more radio-on time, i.e., higher energy consumption. This is unacceptable, especially in mission-critical applications.

To this end, we propose DeCoT, a dependable concurrent transmission based protocol for WSN which works under adverse conditions. Specifically, DeCoT, 1) maintains the connectivity of the network with Scan-and-Lock (as explained in Section III-B) when the links are interfered and 2) runs autonomously with Force-Initiated mechanism (as explained in Section III-C) in the case that the network is partitioned by interference.

We make the following contributions through this work:

- We propose DeCoT, a dependable CT-based WSN to stand against interference.
- We design Scan-and-Lock, a channel hopping mechanism is proposed in DeCoT, to make sure the quality of links under interference.
- We propose Force-Initiated mechanism to enable nodes to be more autonomous in the face of unexpectedly partitioned networks.
- We implement DeCoT in Contiki OS [8] and evaluate the performance under different types of interference in FlockLab [9] and DCUBE [10].

According to our experiments, under the strong interference condition, DeCoT achieves an average reliability of up to 87% with Scan-and-Lock, and outperforms the state-of-the-art flooding protocol Robust Flooding (that won 1st place in the EWSN 2017 Dependability Competition1). Even in a partitioned network, DeCoT can still complete the initialization of the network and works properly due to Force-Initiated mechanism.

DeCoT is able to survive and work normally and efficiently under noise, especially under strong noise. In the EWSN 2018 Dependability Competition2, the average reliability of DeCoT has been evaluated as the best protocol to resist severe interference.

In this paper, we will explain some preliminaries of CT and analyze the consequences of current CT-based networks under interference in Section II. The design of DeCoT is explained in Section III, followed by experiments presented in Section IV. Related work is provided in Section V and Section VI gives brief concluding remarks.

II. PRELIMINARIES

Here, we make a brief introduction of CT to give readers a more comprehensive understanding of CT-based networks. Additionally, we analyze the influence by interference in current CT-based WSNs.

---

1 www.ewsn2017.org/dependability-competition.html
2 ewsn2018.networks.imdea.org/competition-program.html
A. CONCURRENT TRANSMISSION

Concurrent transmission (CT) is defined as that two or more packets are transmitted at exactly the same moment in time. The modulated radio waveforms of packets from several transmitters are superimposed at the receiver. Constructive interference forms and strengthens the superimposed signal if the packets are identical and aligned precisely. For IEEE 802.15.4 receivers, the packets are regarded as being one packet and received correctly as long as the error of the alignment is less than 0.5 $\mu$s [2].

The capture effect - another property of CT - occurs if the alignment of packets is slightly inaccurate. Capture effect is brought by the frequency shift keying (FSK) demodulator, a digital FM demodulator. The stronger signal, which must be at least 3 dB greater than the sum of all the other signals at the receiver, can be demodulated correctly [3]. IEEE 802.15.4 receivers start to record and demodulate only after a valid synchronization header has been detected. To guarantee that the synchronization header of the strongest signal can be properly recognized and decoded by the receiver, the strongest packet must arrive no later than 160 $\mu$s after the weaker packets [5]. The packets with different payloads are able to be overlapped rather than collided due to the capture effect. Specifically, the dominant packet would be received properly and the other packets are overlapped and do not affect the reception.

B. CT-BASED NETWORK

The most representative CT-based protocol of IEEE 802.15.4 is Glossy [2]. As Figure 2 shows, after receiving an identical packet, a node immediately transmits the packet, in order to accurately align this packet with those packets from other transmitters. The back-to-back pattern achieves a precise (less than 0.5 $\mu$s) alignment to propagate a packet without collisions (only three slots are required for three hops in Figure 2). According to the received packets, nodes (N1, N3, N4 and N5) can speculate about the start time of the current period and the next period. Nodes timely turn on radios before the next period starts. All nodes in a CT-based network wake up and sleep synchronously. In terms of the end-to-end performance (i.e., reliability, latency and energy efficiency), Glossy works well in one-to-many scenarios.

For scenarios such as many-to-one and many-to-many, an application-level scheduling is introduced in LWB [4]. The host first initiates the network with scheduling/synchronization packets globally and then sources (i.e., the nodes that have packets to transmit) initiate the flooding in turns. Combined with in-network information processing, Chaos [5] utilizes the capture effect and a bitmap to implement an all-to-all communication. The bitmap records the nodes that have participated in the current all-to-all communication. The nodes remain silent if the received information is known. That reduces the collisions which still exist particularly when too many different packets are transmitted concurrently. Moreover, in Chaos, a timeout mechanism is also introduced to avoid that the process of information exchange halts if unknown packets collide. Specifically, a node re-initiates the network instead of overhearing and remains silent if it has not received a valid packet for a long time.

C. CT-BASED NETWORK UNDER INTERFERENCE

The interference can affect or break the communication link between nodes. Unfortunately, the network is divided into several isolated parts unexpectedly, once several links got broken simultaneously.

1) Lossy Links

The low SNR at the receiver caused by interference leads to unsuccessful receptions. In current CT-based networks, nodes (e.g., node N2 in Figure 2) do not transmit if they receive nothing valid, i.e., they cannot detect the synchronization header or fail to validate the received packet. Meanwhile, the rest of the network (nodes N3, N4 and N5) could still work because of the flooding nature in CT-based networks.
For Chaos, a node does not transmit if it has not been initiated (i.e., has not received a valid packet) in a period. That is to say, in this period, the node does not exchange information at all. This cannot be tolerated in any scenario.

2) Losing Synchronization
In fact, the flooding packets from the host are used to keep nodes woken up and sleeping synchronously, in CT-based networks. Consequently, the nodes lose synchronization with the host if the nodes have not received any valid packet for a long time. The unsynchronized nodes cannot even wake up while other nodes are communicating with each other, no matter whether there exists interference or not. This means that these nodes consume the energy without any contribution of forwarding packets. In LWB or Crystal, it is even worse since an unsynchronized node (like node N2 in Figure 2) cannot initiate the network (i.e., flooding its packet) effectively in its turn, because its neighbors may sleep. As a result, the reliability degrades drastically.

3) Initialization Failure
As mentioned above, CT-based networks rely on a global scheduling of the host. Some nodes work improperly or stay in an uninitialized status when the unexpected partition of the network occurs. As shown in Figure 1, the gray nodes at time T1 are not able to be synchronized with the host because there is interference along the red dashed line. Node N19 is still unable to be synchronized even there exists no interference around it. The reason is that its predecessors N18 and N20 receive nothing due to interference and they do not trigger a transmission to N19. Over time, the interference changes spatially (see time point T2 in Figure 1). The synchronized nodes do not forward packets if they receive nothing. Consequently, the synchronized nodes in the unreachable area (the rightmost at time T2 in Figure 1) do nothing to help nodes N13, N18, N19 and N20 to get synchronized to participate in the network. Nodes in current CT-based networks are only synchronized via the packets from the host. Nodes are required to turn the radio on and off synchronously in CT-based networks. That means unsynchronized/uninitialized nodes cannot propagate packets effectively. It is unfeasible even when node N19 is simply required to exchange information with node N20. More nodes would fail to be initialized, i.e., be synchronized with the host, in a larger networks.

In a word, CT-based networks are faced with the lossy links, losing synchronization and even initialization failure under interference. These obstacles lead to consuming more energy, increasing the end-to-end latency and degrading the reliability.

III. DESIGN
In this section, we present an overview of DeCoT. Then link maintenance with Scan-and-Lock is explained, followed by an explanation of the Force-Initiated mechanism to avoid initialization failure. Moreover, the information exchange is elaborated and some important details of the implementation are presented.

A. DECOT IN A NUTSHELL
Like other current CT-based WSNs, there is a given and fixed host in DeCoT. DeCoT equips itself with a channel hopping strategy. Nodes including the host decide which channel to overhear among the given channels with Scan-and-Lock (i.e., scanning the given channels and locking the clearest channel to overhear) before the start of the period. Once receiving a valid packet, the node forwards it immediately in a back-to-back continuous transmission pattern in different channels. In Force-Initiated periods, synchronized nodes work as synchronization agents and initiate the network (similar to the host). This solves the unexpected network partition problem. A bitmap recording which nodes have participated in the current information exchange, ensures that the unknown information flows efficiently in DeCoT. However, in Force-Initiated periods, the synchronized source nodes (if the source nodes complete the synchronization and work as synchronization agents) initiate the network actively to exchange information rather than wait to be initiated.

B. LINK MAINTENANCE
There are 16 communication channels in the IEEE 802.15.4 standard [7]. If a number of links are affected by interference, the communication can still survive by using a channel hopping mechanism, e.g., nodes can communicate normally on the channel 20 if interference is on the channel 13 only. The more channels are available for channel hopping, the more chance there is to maintain the communication links. However, overhearing on more channels brings about more communication latency and energy consumption led by the increased channel rendezvous time.

Consequently, we do not waste time and energy on overhearing on interfered channels to constrain the channel rendezvous time. The nodes ought to overhear on the...
channel with relatively high quality, because successful receptions can be achieved more probably on a clearer channel. Moreover, the quality of channels has to be profiled precisely. Nodes have to carefully distinguish the noise from collided packets. Any false-positive or false-negative profiling of channel quality results in wrong channel selections and performance degradation.

Based on these two principles together with CT-based networks, a technique named Scan-and-Lock is proposed to make links and flooding robust.

1) Channel Hopping with Scan-and-Lock
As shown in Figure 3, a node scans the received signal strength (RSS) over the given channels $C_1$, $C_2$, to $C_N$ once it wakes up. Then, it locks on the clearest channel to overhear. The host also scans the RSS values over the channels before transmitting the same payload in the order of $C_{i+1}$, $C_{i+2}$, ..., $C_N$, $C_1$, $C_2$, ..., $C_{i-2}$, $C_{i-1}$, $C_i$, if channel $C_i$ is chosen as the channel to listen to. Nodes in CT-based networks turn the radio on to overhear and off to sleep to save energy synchronously. This ensures that the channels with higher RSS values are interfered rather than affected by some collided packets, because no node transmits earlier than the start of the period. After transmitting over all the channels, nodes stay on the channel of the last transmission and receive the packets opportunistically.

2) Best-effort Transmission
To survive under instantaneous interference or collisions, a best-effort transmission (i.e., continuous transmission and re-transmission after a random timeout) is designed in DeCoT. A node transmits continuously and keeps the back-to-back alignment with the other nodes in DeCoT. As shown in Figure 3, if the host receives nothing during random slots which are greater than the amount of the candidate channels, it would re-transmit the packet on the next channel of the overheard one. For the node which has once received a valid packet in this period, the re-transmission mechanism after a random timeout also would be triggered like N1. All the nodes would not stay at the same channel globally (i.e., change channel asynchronously) due to the random timeout mechanism.

C. FORCE-INITIATED MECHANISM
Force-Initiated mechanism in DeCoT solves that some nodes lose synchronization or even fail to setup when the network is partitioned unexpectedly due to the noise mentioned in Section II-C2 and Section II-C3.

The mechanism works as follows. The rounds are divided by the synchronization (Sync) period as shown in Figure 4. There are several nodes work as synchronization agents after they get synchronized with the host. They initiate the CT-based network simultaneously as well as the host in Force-Initiated periods (Force-Initiated periods 1 to $M$ if $M$ periods per round).

As shown in Figure 5, the nodes get synchronized with the synchronization agent (N15 in Figure 5) in the isolated area of the network. The nodes, listening to both the host and the synchronization agent, could get synchronized opportunistically. The asynchronous channel-hopping and the re-transmission in random timeout enable transmissions to be diverse, thereby reducing collisions. To achieve a high dependability, all the nodes in the network can work as synchronization agents.

D. INFORMATION EXCHANGE STRATEGY
In order to support information exchange, DeCoT applies bitmap to enable the exchange. Each source in the network can be represented by a corresponding bit in the bitmap. It is used to record information which has been processed or received. Specifically, ONE in the bitmap represents that the packet is known by the corresponding node. In other words, this node has forwarded or processed the packet. ZERO means that the packet is unknown for the node, i.e., the node has not forwarded or processed the packet. Each node maintains a bitmap by itself. If a received bitmap is different from that maintained by the receiver, the receiver processes/updates the payload of the packet, ORs its bitmap.
with the received bitmap, and forwards it. The receiver transmits nothing if the received bitmap is identical with that in the receiver.

In Force-Initiated periods, synchronization agents initiate the network simultaneously. That is to say, the information exchange can start from everywhere at the same moment. Therefore, it completes more quickly if sources are sparse enough.

E. IMPLEMENTATION ASPECTS
We implement DeCoT based on Contiki OS on Tmote Sky [11] and we have three points to present on the aspect of implementation.

Guard Time: On the one hand, Scan-and-Lock is executed immediately once the node wakes up. We must guarantee there are no nodes transmitting during scanning. On the other hand, Force-Initiated mechanism brings about the inaccuracy of the clock synchronization. Thus, the guard time of waking up in DeCoT has to be designed dedicatedly. After our experiments in a network with 51 nodes and a diameter of 5 hops, we find 5 ms is enough for scanning 6 channels.

Energy Optimization after Accomplishment of Information Exchange: Once a node has received a packet with the information of all the sources, it means there is no unknown information for this node. In this case, the node only repeats transmitting the packet in random slots to guarantee the reliability. To save energy, it turns the radio off rather than overhears after transmission.

Random Channel Sequence of Continuous Transmission: The collision occurs more probably because the alignments of continuous transmissions of different nodes could not be as accurate as the receive-and-transmit pattern. For example, in Figure 3, a collision would occur probably at node N4 on the channel C2 due to the inaccurate alignment of N1 and N2. To relieve such a collision, DeCoT uses a random sequence for selecting channels and this sequence is updated after each period.

IV. EXPERIMENTS
We conduct experiments in two testbeds: FlockLab [9] in ETH and DCUBE [10] [12] in TU Graz. There are 31 Tmote Sky nodes totally (4 are outdoor and 27 are indoor) in FlockLab. Each node is monitored by an observer to record some pins and serial messages of the node. DCUBE consists of 51 Tmote Sky nodes over multiple floors in the area of 1000 m². Unlike FlockLab, DCUBE is dedicated to the EWSN 2018 Dependability Competition and is only accessible during the preparation phase (from Dec., 2017 to Jan., 2018). Each Tmote Sky node in DCUBE is equipped with an interference generator which is based on the WiFi NIC of Raspberry PI. The Raspberry PI also monitors the nodes like the observer in FlockLab.

In FlockLab: To evaluate the performance of flooding with the channel-hopping mechanism, we compare DeCoT Flooding (DeCoT without the information exchange mechanism and Force-Initiated mechanism) with Glossy [2] and the Robust Flooding (RoF) [13] under interference. Then, in order to figure out whether the continuous transmission, Scan-and-Lock and Force-Initiated mechanism would affect information exchange like all-to-all communication, we run Chaos [5]. DeCoT and Chaos with Continuous Transmission (Con. Tx Chaos). After that, we construct a partitioned network by noise to test whether DeCoT is able to complete the initialization and work properly. Crystal3 (the dependability competition version based on [6]), Con. Tx Chaos and DeCoT Flooding are also tested.

In DCUBE: Besides, we show the evaluation of DeCoT under different interference levels in the EWSN 2018 Dependability Competition.

A. FLOODING PERFORMANCE OF DECOT
We evaluate the flooding performance in terms of end-to-end (E2E) latency, E2E reliability, flooding reliability and average energy consumption.

1) Setups
We use 27 indoor nodes of FlockLab and 7 of them (node 4, 33, 32, 28, 10, 19 and 13) work as interferers to make the interference cover the network as possible as we can, as shown in Figure 6. Packets are required to be propagated from node 1 to node 7 and other 18 nodes work as relays. To guarantee that the interferers are able to affect the network, interferers create noise on variable channels with the maximum power (0 dBm) and other nodes communicate at -5 dBm. We do not use the outdoor nodes due to the poor connectivity at -5 dBm.

The noise is generated by JamLab [14]. To achieve noise in a broader band, interferers control the wireless transceiver

3 iti-testbed.tugraz.at
4 github.com/d3s-trento/crystal/tree/depcomp18
CC2420 on Tmote Sky to transmit the modulated wireless signal rather than the unmodulated one as noise. As listed in Table 1, five types of noise are designed. For example, in Noise 1, all the seven interferers (as mentioned before, yellow nodes in Figure 6) generate the noise with the maximum transmit power, 0 dBm. They switch to a random channel among channel 11, 15, and 26 every 0.5 s. In addition to the noise with constant transmit power and variable channels (Noise 1 to 4), the noise on fixed channels with variable transmit power (i.e., Noise 5) is also applied to our experiments. It is notable that the amount of the interferers is constant, i.e., 7, in all 5 types of noise. That is to say, Noise 1 is the most intensive in those three given channels and Noise 5 is the weakest.

In the flooding performance evaluation, the original Glossy, the state-of-the-art dependable flooding protocol RoF, and DeCoT Flooding with different configurations are tested. RoF is a flooding protocol based on continuous transmission with channel hopping. However, RoF does not rely on channel scan. The channel, on which the node can receive the packet successfully, is added to a list. The node in RoF picks a channel to overhear randomly from the list. The full configurations of the protocols are listed in Table 2. The duration (i.e., the radio-on time to overhear) and the period (i.e., period of waking up) of all the flooding protocols are 147 ms and 200 ms, respectively. As mentioned above, the transmit power is set to -5 dBm.

2) Evaluations

We run Glossy in a clear environment and 5 scenarios with various settings of noise. RoF2_3ch and DeCoT Flooding_3ch are tested without noise. For the experiments of channel hopping flooding protocols (RoF and DeCoT Flooding), we test them under the type of noise with the corresponding interfered channels. For example, RoF1_4ch, RoF2_4ch and DeCoT Flooding_4ch are tested under Noise 2 because Noise 2 interferers on the 4 channels. Interferers of Noise 5 disturb on the 3 given channels as the ones in Noise 1, thus RoF1_3ch, RoF2_3ch and DeCoT Flooding_3ch are also tested under Noise 5. Five experiments are conducted for each protocol under the corresponding noise. More than 2000 packets of flooding are computed.

Four criteria are used to profile the performance of flooding: average E2E latency, average E2E reliability, average flooding reliability and average energy consumption.

Average E2E latency: Average E2E latency is the time from the host (node 1 in FlockLab) transmitting the first packet to the sink (node 7 in FlockLab) first receiving successfully in each flooding period. To eliminate the difference brought by the length of default flooding packet, we use the number of slots to compute the E2E latency.

Average E2E reliability: Average E2E reliability is the Packets Delivery Rate (PDR) from the host (the node 1 in FlockLab) to the sink (the node 7 in FlockLab), i.e., the ratio of the number of packets received by the sink to that transmitted by the host.

Average flooding reliability: Flooding reliability is represented by average PDRs of all the nodes. Besides the E2E reliability, average flooding reliability is another criterion since an efficient information exchange like many-to-many communication relies on the high flooding reliability.

Average energy consumption: Energy consumption is an important requirement in practical low-power communication systems. Measuring accurate energy consumption is difficult since many operations like the output for logging and the LEDs for debugging consume more energy. For CT-based system, average radio-on time could be used to profile the power consumption. We calculate the average radio-on time per byte to avoid the influence of different lengths of packets in protocols.

3) Results

E2E latency: In FlockLab, the mean of the latency without interference of the original Glossy is the time of 3.4 slots and the standard deviation is 0.5 slots when the transmit power is -5 dBm according to our 5 experiments. All the E2E latencies of other protocols with channel hopping are above 3.4 slots due to the rendezvous time as shown in Figure 7. Under the most intensive interference (Noise 1), the E2E latency of DeCoT Flooding and RoF_1 are both 8.6 slots, but DeCoT Flooding is more stable (with a smaller standard deviation). For DeCoT Flooding, more channel hopping rendezvous time leads to more E2E latency which is apparent from Noise 1 to 4. In contrast, RoF does well in latency when the noise is weaker (the Noise 2 to 5) since nodes in RoF change channel more frequently and randomly than Scan-and-Lock in DeCoT Flooding. In other words, frequent changing channels to overhear increases the receiving chances in the channel hopping system and reduces the rendezvous time. Particularly, more channels are available under the weaker noise. But according to the E2E latency of DeCoT Flooding under Noise 1 and 5, the latency in DeCoT Flooding does not increase when noise becomes intensive.

E2E reliability and flooding reliability: The corresponding reliabilities under different types of noise and a none-
receive nothing during the whole duration and it is energy-consuming. As shown in Figure 9, receiving-nothing rounds waste plentiful energy and the RoF1_3ch even consumes the energy as much as RoF2_3ch under Noise 1. Similarly, the average energy consumption of RoF1_3ch is much more than RoF2_3ch under the Noise 5 due to the receiving-nothing rounds. However, the Scan-and-Lock mechanism avails the nodes in DeCoT to lock the clear or less-interfered channel and reduce the amount of receiving-nothing rounds to save energy.

From these experiments, we see DeCoT Flooding outperforms the original Glossy and RoF under the intensive interference. When the interference becomes weak, RoF performs better in E2E latency.

B. INFORMATION EXCHANGE PERFORMANCE OF DECO T

Through the flooding evaluation without information exchange, we know that DeCoT is able to escape from the interference with Scan-and-Lock to achieve a dependable performance. Now, we need to know whether the continuous transmission with Scan-and-Lock and Force-Initiated mechanism affect the information exchange.

1) Setups

26 indoor nodes (except node 17 since it is not always accessible during the experiments) in FlockLab are used to evaluate the information exchange performance. Outdoor nodes are not used since the connectivity cannot be guaranteed when the transmit power is -5 dBm.

Chaos, Con. Tx Chaos and DeCoT are evaluated. Con. Tx Chaos is a version of Chaos with continuous transmission and Scan-and-Lock. The number of hopping channels and the number of continuous transmissions are both 3. For DeCoT, the sources but not the host work as synchronization agents.

interfered scenario are illustrated in Figure 8. The channel-hopping mechanisms of RoF and DeCoT Flooding both improve the reliability significantly. The E2E and flooding reliability of DeCoT Flooding are about 87% and 87.9%, respectively even under the most intensive noise (Noise 1). DeCoT Flooding is better than RoF apparently under intensive noise (Noise 1 and 2) according to Figure 8. The reliabilities of the original Glossy under various noise in Figure 8 also reflect the intensity of the interference as our expectation (Noise 1 is the strongest and Noise 5 is the weakest).

Average energy consumption: In terms of energy, DeCoT Flooding also performs better under stronger interference. Since the random channel picking strategy of RoF could not select a proper clear channel, the nodes in RoF would

![TABLE 1: Configurations of interferers in the flooding performance evaluation.](image1)

<table>
<thead>
<tr>
<th>Noise 1</th>
<th>Noise 2</th>
<th>Noise 3</th>
<th>Noise 4</th>
<th>Noise 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 dBm</td>
<td>0 dBm</td>
<td>0 dBm</td>
<td>0 dBm</td>
<td>alter from -25 dBm to 0 dBm randomly per 0.5 s</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Switch randomly per 0.5 s</th>
<th>Switch randomly per 0.5 s</th>
<th>Switch randomly per 0.5 s</th>
<th>Switch randomly per 0.5 s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noise 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Noise 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Noise 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Noise 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Noise 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![TABLE 2: Configurations of flooding protocols in the flooding performance evaluation.](image2)

<table>
<thead>
<tr>
<th>Original Glossy</th>
<th>RoF1_3ch</th>
<th>RoF1_4ch</th>
<th>RoF1_5ch</th>
<th>RoF1_6ch</th>
<th>RoF2_3ch</th>
<th>RoF2_4ch</th>
<th>RoF2_5ch</th>
<th>RoF2_6ch</th>
<th>DeCoT Flooding_3ch</th>
<th>DeCoT Flooding_4ch</th>
<th>DeCoT Flooding_5ch</th>
<th>DeCoT Flooding_6ch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. Num. Transmissions</td>
<td>7</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Num. of Hopping Channels</td>
<td>1 (no hopping)</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>Channels</td>
<td>1, 11, 15, 26</td>
<td>1, 11, 15, 20, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
<td>1, 11, 15, 20, 23, 26</td>
</tr>
<tr>
<td>Num. of Continuous Transmissions</td>
<td>1 (no continuous transmission)</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>Packet Length (Byte)</td>
<td>9</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

![Figure 7: The end-to-end latency under interference.](image3)
Therefore, there are 25 synchronization agents in the all-to-all scenario in DeCoT. For DeCoT, 3, 5 and 7 Force-Initiated periods per round are all tested. The node 1 in Flocklab (in Figure 6) is the host of all the tested protocols. The protocols are tested in a dense network (transmit power is 0 dBm) and a sparse network (transmit power is -5 dBm) respectively.

2) Evaluations

Two scenarios which are common in the practical applications are considered to profile the information exchange performance: all-to-all and many-to-all communication. In all-to-all communication, every node works as a source node. The information generated by a source node needs to be propagated to the whole network. In many-to-all communication, where not all nodes generate information, some of them work as normal relays and the information from sources need to be propagated to all the nodes. These selected source nodes are expected distributed in the whole network uniformly. Specifically, we test 8 nodes (node 1, 8, 32, 6, 22, 24, 20 and 7 in FlockLab) to all and 16 nodes (node 1, 2, 8, 15, 32, 3, 6, 16, 22, 27, 24, 26, 20, 25, 14 and 7 in Flocklab) to all respectively. Similar to the evaluations of the flooding performance, each experiment is repeated for 5 times. More than 2000 packets of flooding are computed.

In the evaluation, we use two main metrics, i.e., reliability and efficiency, to address the performance of different protocols.

**Average information exchange reliability:** In the all-to-all evaluation, average information exchange reliability is obtained by all-to-all communication success rate, i.e., the success rate of one node receiving all the information in one period. In many-to-all evaluation, it refers to many-to-all communication success rate, i.e., the success rate of one node receiving all the information from the sources in one period.

**Average information exchange efficiency:** It is notable that there is processing time (4000 MCU cycles of Tmote Sky in default configuration, about 10 ms) between receiving and transmitting in Chaos. However, there is no processing time in Con. Tx Chaos and DeCoT because in DeCoT, we read/write data from/to the radio transceiver with a faster manner, the direct memory access (DMA). Thus we also use number of slots to compute the speed of completing information exchange. In all-to-all evaluation, average information exchange efficiency is defined as the number of slots to complete an all-to-all communication per period. Similarly, in many-to-all evaluation, it is represented by the amount of slots used to achieve a many-to-all communication.
3) Results

All-to-all performance: The average all-to-all communication reliability of all the protocols are more than 99.7% as shown in Figure 10a. However, in Figure 10b, the all-to-all efficiency of the original Chaos is higher than Con. Tx Chaos. It is more apparent when the network becomes dense (the transmit power is 0 dBm). Chaos spends about 25 slots to complete all-to-all communication, but Con. Tx Chaos and DeCoT still need more than 50 slots. Two factors lead to these results: rendezvous time of channel hopping and continuous transmission. They both lengthen the time of each information exchange (receive-and-transmit) operation thereby reduce the information exchange efficiency.

For the sparse network, in Force-Initiated periods, DeCoT reduces the amount of slots to complete all-to-all communication since Force-Initiated mechanism enables an all-to-all communication to start everywhere rather than start from the host. Thus, more Force-Initiated periods in one round bring about less average slots to accomplish all-to-all communication (i.e., high efficiency of all-to-all communication) as shown in Figure 10b. DeCoT with 7 Force-Initiated periods is the most efficient all-to-all communication based on the continuous transmission in the sparse network in our experiments.

Nevertheless, DeCoT does not improve the efficiency in dense network. With the increment of the Force-Initiated periods per round, the amount of slots to complete all-to-all communication increases as shown in Figure 10b. This is because the nodes which initiate the network in Force-Initiated periods are too dense and they can hardly receive valid packets at the beginning of the Force-Initiated periods. First slots of Force-Initiated periods are used to force the network to be initiated. Afterwards, some nodes start to overhear and others re-transmit due to the random timeout mechanism. Only after that, the effective information exchanges start.

Many-to-all performance: The reliabilities of the 16-to-all and the 8-to-all are both more than 99.8% as shown in Figure 11a and 12a.

Similar to the scenario of all-to-all communication with 0 dBm, the efficiencies of DeCoT are not as good as that of Con. Tx Chaos since the sources are dense and serious collisions happen at the beginning of Force-Initiated periods. However, in the scenario of 8-to-all, Force-Initiated mechanism reduces the slots used to accomplish 8-to-all even with 0 dBm because the selected 8 source nodes in FlockLab are sparse for DeCoT.

After these experiments, Chaos, Con. Tx Chaos and DeCoT all reach high information exchange reliability. However, the continuous transmission with channel hopping decreases the information exchange efficiency. But we also find that DeCoT can improve it when the sources are placed sparsely.

C. DECOT IN PARTITIONED NETWORK

1) Setups

We construct a partitioned network with interference in FlockLab. Specifically, node 33, 31, 28, 10, 19 and 13 are assigned as interferers. The network is divided into left and right parts and both two parts do not keep silent at the same moment as illustrated in Figure 13. Namely, the left three interferers work when the right three keep silent and vice versa. To guarantee the network could be partitioned, the transmit powers of interferers are set to the maximum (0 dBm). Thus, all the interferers work only at one given channel and all the tested protocols are also configured to work at the same channel (i.e., channel 26) without channel hopping. To observe whether these tested protocols could complete their initializations even when the period of disabled interferer (i.e., the gray part in Figure 13) becomes longer, two kinds of partitioned networks are designed as listed in Table 3. The interval in Table 3 is the time from T1 to T2 in Figure 13. Similar to Section IV-A, all the interference is modulated and generated by JamLab.

The host in all the tested protocols is set to node 1 in FlockLab and the payload of a packet is set to 1 Byte. Besides, there are 4 pairs of one-to-one communications in this scenario, i.e., packets from node 1, node 18, node 7, and node 16 are required to be received by node 15, node 24, node 14, and node 11, respectively. The Max. Tx number in Crystal (the dependability competition version) is set to 5. Con. Tx Chaos runs with 3 continuous transmissions. The amounts of continuous transmissions in DeCoT and DeCoT Flooding are also set to 3. The Max. Tx numbers of DeCoT and DeCoT Flooding are 6. Crystal runs with the periods of 500 ms (since we find the print_stats(), a statistic output function in its code, could not work properly in smaller periods) and others run with the periods of 200 ms. The transmit power of nodes is set to -3 dBm. In DeCoT, as the evaluations of information exchange, all the sources but not the host work as synchronization agents, i.e., there are 19 synchronization agents.

2) Evaluations

We run Crystal, Con. Tx Chaos, DeCoT and DeCoT Flooding to observe how they perform when the network is partitioned unexpectedly by the interference. DeCoT Flooding, a flooding protocol which cannot achieve the 4 pairs of one-to-one communication, is used to observe whether it could complete the initialization. Each protocol is repeated for 5 times in both partitioned networks respectively and each test lasts 10 minutes. The average E2E reliabilities are computed if the network completes the initialization.

3) Results

Crystal, Con. Tx Chaos and DeCoT Flooding fail to initialize: Crystal, Con. Tx Chaos and DeCoT Flooding are not able to finish the initialization of the whole network at all. The initializations of experiments are illustrated in Figure 14a, 14b, 14c, 15a, 15b and 15c. There are 14 nodes out of 20 nodes that could complete the initialization in
FIGURE 10: All-to-all communication performance. DeCoT \( n \) FI means DeCoT with \( n \) Force-Initiated Periods.

FIGURE 11: 16-to-all communication performance.

FIGURE 12: 8-to-all communication performance.
most experiments, as shown in Figure 15b. One exception is the experiment of Crystal in the scenario of the partitioned network 2. 15 nodes is initialized as shown in Figure 15b. It is the unexpected partitioned network that always leaves these nodes (node 11, 14, 7, 25, 20 and 26) uninitialized. The synchronization packets or flooding packets from the host (node 1) cannot reach node 11, 14, 7, 25, 20 and 26 when the right-side interferers are silent because the left-side interferers still work. These nodes receive nothing when the right interferers work, even at the moment the left interferes become silent and sync/flooding packets from the host could reach node 23 or 24. In fact, these nodes still have not been initialized at the end of these experiments. They have to overhear most time to get synchronized and do not forward packets.

**DeCoT works:** It is because of the Force-Initiated mechanism that DeCoT could complete initialization in 20 s mostly (the worst case is in 30 s) even in the partitioned networks as shown in Figure 14d, 14e, 14f, 15d, 15e and 15f. Node 11, 14, 7, 25, 20 and 26 can receive packets from node 23 or 24 when the right-side interferers are silent. The nodes in the left part like node 23 and 24 which have been initialized also initiate the network in the Force-Initiated periods. It is independent of whether they receive the packet from the host or not.

The average E2E reliabilities are shown in Figure 16. The E2E reliability of node 15 receiving packets from node 1 is close to 100% since the interferer 33 could hardly interferes node 15. Actually, the average percentages of period of disabled interferer in the partitioned network 1 and 2 are only about 20% (1.25/6) and 31% (2.5/8) respectively. However, the interferers do not disturb node 24, 14 and 11. Therefore, the node 24, 14 and 11 all achieve reliabilities of more than 60%. In Sync periods of DeCoT, node 18, 16 and 7 would receive nothing probably in the partitioned network because only the host (node 1) initiates the network. Consequently, the more Force-Initiated periods per round DeCoT uses, the less receiving-nothing periods there are. Higher average E2E reliabilities are achieved when the amount of Force-Initiated periods per round increases.

Through the experiments in the partitioned network, traditional CT-based protocols are found to hardly complete

---

**FIGURE 13:** The unexpected partitioned network is constructed by the interferers in FlockLab. Node 17 in black was not accessible during experiments.

**TABLE 3:** Configuration of interference to construct the unexpected partitioned network.

<table>
<thead>
<tr>
<th>Partitioned Network</th>
<th>Period of Disabled Interferer (s)</th>
<th>Channel of Interference</th>
<th>Power of Interference</th>
<th>Interval (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network 1</td>
<td>random number in [1, 1.5]</td>
<td>26</td>
<td>0 dBm</td>
<td>6</td>
</tr>
<tr>
<td>Network 2</td>
<td>random number in [2, 3]</td>
<td>26</td>
<td>0 dBm</td>
<td>8</td>
</tr>
</tbody>
</table>
FIGURE 14: Initializations of tested protocols in the partitioned network 1.

FIGURE 15: Initializations of tested protocols in the partitioned network 2.
the initialization of the whole network but DeCoT is able to overcome this challenge.

D. EVALUATION OF DECOT IN DCUBE

An original version of DeCoT (eOFPCOIN [15]) was also tested in DCUBE and evaluated in the EWSN 2018 Dependability Competition.

1) Setups

In the EWSN 2018 Dependability Competition, many source nodes monitor several events (like the transitions of GPIO) and need to forward this information to one or more destinations within a multi-hop network. There are three kinds of communications: point-to-point, point-to-multipoint and multipoint-to-point.

Point-to-point: The source sends the monitored event to a destination and the destination triggers the corresponding GPIO.

Point-to-multipoint: The source sends the monitored event to two or more destinations and they trigger the corresponding GPIO.

Multipoint-to-point: Two or more nodes send the monitored events to a destination and it ORs the events from the sources and triggers the corresponding GPIO.

The final evaluation scenario consists of three point-to-point, three point-to-multipoint and two multipoint-to-point communications. Among the 51 nodes, there are 11 sources, 13 destinations and 27 relays.

As Table 4 shows, the available types of interference are different during the preparation phase and in the final evaluation. The interference channels are unknown. The network would be partitioned unexpectedly probably under IP3. The noise used in the final evaluation is a combination of IPx spatially.

DeCoT tested in DCUBE hops in the three channels (channel “10.5”, 15 and 26). Channel “10.5” is at 2.5 MHz smaller than channel 11. They are found to work well under WiFi interference. The duration is 143 ms and the period is 200 ms. There are five Force-Initiated periods per round. The number of continuous transmissions is also three and the Max. Tx number is 90 to guarantee the information exchange reliability which is important especially for the multipoint-to-point scenarios. The transmit power is 0 dBm. The 11 sources defined in the competition work as synchronization agents and initiate the network simultaneously in Force-Initiated periods. According to the requirements of the competition, the payload of DeCoT contains all the monitored events, i.e., status of GPIOs of the source nodes. Information of events are expected to be exchanged in each period.

2) Evaluations

The end-to-end reliability refers to the reliability of events rather than packets. The end-to-end latency is the interval between the moment when an event occurs and the time when the event is triggered by the destination. During the preparation phase, DeCoT is tested under IP1 to IP3 for 3 times. Each time lasts 5 minutes (the maximum test time during the preparation phase). In the final evaluation, DeCoT is evaluated by the official organizer and run under IC0 to IC4 for 3 times, 30 minutes per time, IC5 to IC6 for 5 times, 70 minutes per time.

3) Results

The results are listed in Table 5. The average power per node shows that the less average E2E reliability is achieved, the more energy is consumed. This is similar to the evaluations of DeCoT Flooding (i.e., receiving-nothing period makes energy waste). That is to say, DeCoT consumes more power under more intensive interference and less under weaker noise or clear environment.

In the final official evaluations, DeCoT is the best in terms of the average E2E reliability, particularly under the severest interference. In the universal evaluations, where the total energy consumption, the average E2E reliability and latency are all considered, DeCoT won the third place.

E. SUMMARY

Through experiments, in terms of reliability, DeCoT performs well under interference, especially intensive interference. However, DeCoT still could not be defined as a totally distributed CT-based network because the given and fixed host is still required. Obviously, DeCoT could not face the situation that the host is isolated totally (i.e., there are no nodes that could hear the packets the host transmits). The flooding latency and information exchange efficiency also need to be improved to make DeCoT more resilient. Synchronization agents in Force-Initiated mechanism enable DeCoT to survive under the interference partitioning network at the cost of information exchange efficiency. However, a proper...
synchronization agents selection can improve information exchange efficiency, particularly in dense network.

V. RELATED WORK
In this section, we survey the state-of-the-art protocols that provide reliable and robust CT-based communications for multi-hop low-power WSNs.

Glossy [2], proposed by Ferrari et al. in 2011, provides a fast and efficient network flooding service by using concurrent transmissions in WSNs. By exploiting constructive interference and capture effect on physical layer, Glossy achieves an average packet delivery ratio of 99.99% and ultra low latency in real testbeds. However, Glossy aims to provide reliable and robust CT-based communications for one-to-many applications. Thus it is not applicable for many-to-one applications such as data collection. To realize the design of many-to-one application with Glossy, Ferrari et al. add an application-level scheduler to construct a so-called Low-power Wireless Bus (LWB) [4]. LWB centrally schedules the data communication to support one-to-many, many-to-one, and many-to-many traffic patterns in WSNs.

Later, Splash [16] builds a tree pipeline [17] by exploiting Glossy, thereby improving channel utilization. Furthermore, Pando [18] integrates fountain code with pipeline to overcome the long-tail problem of Splash. While Glossy disseminates one packet in each communication round, Splash and Pando are designed to deliver large data objects to all nodes in a network, e.g., for the purpose of reprogramming the WSN-based applications. Ripple [19] also relies on Splash and network coding techniques to improve particularly in terms of network throughput.

More recently, Chaos [5] builds on Glossy to achieve fast all-to-all data sharing in a distributed manner. Chaos further combines programmable in-network processing with concurrent transmissions in WSNs. However, Chaos performs data dissemination in parallel by integrating an aggregate function into concurrent transmission schedules, such as MAX, MIN, COUNT, and so forth. On the contrary, Codecast [20] provides a more general approach to support many-to-many communication. It introduces a feedback-driven network coding (NANC) into synchronous transmission schedules. According to the evaluations, this scheme achieves reliable and high throughput many-to-many data sharing.

For data collection (many-to-one) scenarios, Crystal [6] makes the scheduling simpler than LWB by TA pairs, where T represents a data transmission slot and A is an acknowledgement slot. Sources transmit packets simultaneously, which would be received by the sink opportunistically, in a T period. Then, the sink floods an acknowledgement packet to tell all the nodes which are received in A period. Sources transmitting simultaneously in T is similar to DeCoT. Different with DeCoT, it is not used to synchronize in Crystal. This is why Crystal, even with channel hopping scheme [21], still cannot survive when network is partitioned unexpectedly. Crystal with channel hopping scheme introduces a noise detection scheme, that the RF transceiver is turned off when

![Image](http://www.ieee.org/publications_standards/publications/rights/index.html for more information.)

### TABLE 4: Types of interference in DCUBE

<table>
<thead>
<tr>
<th>Interference</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP1 (Interference 1 during Preparation)</td>
<td>Only on a single frequency</td>
</tr>
<tr>
<td>IP2 (Interference 2 during Preparation)</td>
<td>On multiple frequencies (mild)</td>
</tr>
<tr>
<td>IP3 (Interference 3 during Preparation)</td>
<td>On multiple frequencies (strong)</td>
</tr>
<tr>
<td>IC0 (Interference 0 in Final Competition)</td>
<td>Absence of Interference</td>
</tr>
<tr>
<td>IC1 (Interference 1 in Final Competition)</td>
<td>Bursts of fixed duration, same fixed channel for all interferers (mild)</td>
</tr>
<tr>
<td>IC2 (Interference 2 in Final Competition)</td>
<td>Bursts of fixed duration, fixed random channel for all interferers (mild)</td>
</tr>
<tr>
<td>IC3 (Interference 3 in Final Competition)</td>
<td>Bursts of varying duration, fixed random channel for all interferers (medium)</td>
</tr>
<tr>
<td>IC4 (Interference 4 in Final Competition)</td>
<td>Bursts of fixed duration, dynamic channel for all interferers (mild)</td>
</tr>
<tr>
<td>IC5 (Interference 5 in Final Competition)</td>
<td>Bursts of varying duration, dynamic channel for all interferers (medium)</td>
</tr>
<tr>
<td>IC6 (Interference 6 in Final Competition)</td>
<td>Bursts of varying duration, dynamic channel for all interferers (strong)</td>
</tr>
</tbody>
</table>

### TABLE 5: Results in DCUBE

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IP1</td>
<td>99.4</td>
<td>264.1</td>
<td>1190.1</td>
</tr>
<tr>
<td>IP2</td>
<td>99.2</td>
<td>267.7</td>
<td>1142.3</td>
</tr>
<tr>
<td>IP3</td>
<td>66.9</td>
<td>889.3</td>
<td>1302.1</td>
</tr>
<tr>
<td>IC0</td>
<td>99.63</td>
<td>242.63</td>
<td>5631</td>
</tr>
<tr>
<td>IC1</td>
<td>99.11</td>
<td>275.72</td>
<td>7041</td>
</tr>
<tr>
<td>IC2</td>
<td>97.25</td>
<td>336.19</td>
<td>7258</td>
</tr>
<tr>
<td>IC3</td>
<td>80.36</td>
<td>384.05</td>
<td>6823</td>
</tr>
<tr>
<td>IC4</td>
<td>97.42</td>
<td>296.94</td>
<td>7198</td>
</tr>
<tr>
<td>IC5</td>
<td>80.09</td>
<td>347.28</td>
<td>15805</td>
</tr>
<tr>
<td>IC6</td>
<td>80.11</td>
<td>393.85</td>
<td>16718</td>
</tr>
</tbody>
</table>
the interference is too intensive and the node has nothing to transmit, to save energy.

However, a well-designed protocol still cannot meet all application requirements. To improve reliability, similar to the multichannel communication (e.g., MicMAC [22] and MOR [23]) in asynchronous networks, a large number of channel-hopping CT-based protocols [13], [15], [24]–[28] are proposed in the recent EWSN Dependability Competitions. The Robust Flooding [13], is the most similar to DeCoT. It also uses continuous transmission and channel-hopping mechanism which is more randomly than Scan-and-Lock in DeCoT.

Different from the state-of-the-art protocols, DeCoT exploits the synchrony of the energy saving mechanism in CT-based networks to scan channels and lock to the best one, which is called Scan-and-Lock scheme. Under interference, the rendezvous time of nodes can be decreased significantly which results a higher energy efficiency of the network. Besides, an initialization/synchronization assisted scheme (Force-Initiated mechanism) is also introduced in DeCoT in order to ensure a reliable network initialization and to resist adverse interference. According to our experimental results, DeCoT is able to achieve highly dependable performance and outperforms the state-of-the-art protocols especially under severe interference.

VI. CONCLUSIONS

DeCoT, as a dependable CT-based protocol for WSN, is designed to maintain the communication under interference with continuous transmission and Scan-and-Lock. Nodes with Scan-and-Lock can profile the channel exactly and lock to the clearest channel (with the highest reception success rate generally) to overhear. Based on our experiments, DeCoT achieves high dependability of links and outperforms the tested protocols especially under intensive interference. Synchronization agents initiate the network simultaneously in Force-Initiated periods. Consequently, DeCoT can complete the initialization and work properly even when the network is partitioned unexpectedly by interference. This is proven by our experiments and the evaluation of the EWSN 2018 Dependability Competition. In a word, DeCoT can survive and work dependably under interference, particularly strong interference.

ACKNOWLEDGMENT

We would like to thank the Computer Engineering Group at ETH Zürich for providing the FlockLab testbed, as well as the authors of DCUBE testbed from TU Graz. We also thank Ye Liu and Pei Tian for their valuable comments.

REFERENCES


Xiaoyuan Ma received the B.S. degree in 2010 and M.S. degree in 2014, respectively, both from Shanghai Jiao Tong University, Shanghai, P. R. China. He joined the Safety and Emergency Laboratory, Shanghai Advanced Research Institute, Chinese Academy of Sciences (http://english.sari.cas.cn/) in 2011. Now he is pursuing his doctor degree in Shanghai Advanced Research Institute, CAS. His research interests include wireless sensor networks, implementations of embedded system.

Peilin Zhang is a researcher in the System Software and Distributed Systems (SVS) group at the Carl von Ossietzky University of Oldenburg, Germany. He received his M.Sc. in Communications and Systems Engineering at University of Vaasa, Finland, in 2014. His research interests are related to QoS provisions, low-power wireless systems, and wireless sensor networks.

Xin Li is a Ph.D. student in School of Information Science and Technology of ShanghaiTech University. He works in Safety and Emergency Laboratory, Shanghai Advanced Research Institute, Chinese Academy of Sciences (http://english.sari.cas.cn/). His research interests include wireless sensor networks, cross-tech communication and backscatter communication.

Weisheng Tang received the B.S. degree in Electronics and Information Engineering from Central South University, Changsha, China in 2011 and the M.S. degree in Electronics and Communication Engineering from Shanghai Advanced Research Institute, Chinese Academy of Sciences in 2015. He is currently a Ph.D. student in Communication and Information Systems at Shanghai Advanced Research Institute, Chinese Academy of Sciences. His research interests include wireless sensor networks, routing protocols and mobile computing.

Jianming Wei received the Ph.D. degree in communication and information system at Shanghai University, Shanghai, China, in 2005. His research interests include the design of wireless sensor networks for signal processing and communication systems. He is currently working on the system design for collaborative processing and information fusion as well as the application in emergency rescue for public safety.

Oliver Theel is a Professor in the Department of Computer Science, Carl von Ossietzky University of Oldenburg, Germany, where he is heading the System Software and Distributed Systems (SVS) group. His research interests are related to distributed systems, dependable computer systems, fault-tolerant computing systems, self-stabilizing algorithms, region-adherent algorithms and wireless sensor networks.

***