IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 25, 2021, accepted April 15, 2021, date of publication April 21, 2021, date of current version April 29, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3074594

A Dynamic Distributed Multi-Channel TDMA Slot Management Protocol for Ad Hoc Networks

IRFAN JABANDŽIĆ[1], SPILIOS GIANNOULIS[1], (Member, IEEE), RUBEN MENNES[2], (Associate Member, IEEE), FELIPE A. P. DE FIGUEIREDO[1], MAXIM CLAEYS[2], AND INGRID MOERMAN[1], (Member, IEEE)
[1]IDLab, Department of Information Technology, Ghent University—IMEC, 9052 Ghent, Belgium
[2]IDLab, Department of Information Technology, University of Antwerp—IMEC, 2000 Antwerp, Belgium
Corresponding author: Irfan Jabandžić (irfan.jabandzic@ugent.be)

ABSTRACT With the emergence of new technologies and standards for wireless communications and an increase in application and user requirements, the number and density of deployed wireless ad hoc networks is increasing. For deterministic ad hoc networks, Time-Division Multiple Access (TDMA) is a popular medium access scheme, with many distributed TDMA scheduling algorithms being proposed. However, with increasing traffic demands and the number of wireless devices, proposed protocols are facing scalability issues. Besides, these protocols are achieving suboptimal spatial spectrum reuse as a result of the unsolved exposed node problem. Due to a shortage of available spectrum, a shift from fixed spectrum allocation to more dynamic spectrum sharing is anticipated. For dynamic spectrum sharing, improved distributed scheduling protocols are needed to increase spectral efficiency and support the coexistence of multiple co-located networks. Hence, in this paper, we propose a dynamic distributed multi-channel TDMA (DDMC-TDMA) slot management protocol based on control messages exchanged between one-hop network neighbors and execution of slot allocation and removal procedures between sender and receiver nodes. DDMC-TDMA is a topology-agnostic slot management protocol suitable for large-scale and high-density ad hoc networks. The performance of DDMC-TDMA has been evaluated for various topologies and scenarios in the ns-3 simulator. Simulation results indicate that DDMC-TDMA offers near-optimal spectrum utilization by solving both hidden and exposed node problems. Moreover, it proves to be a highly scalable protocol, showing no performance degradation for large-scale and high-density networks and achieving coexistence with unknown wireless networks operating in the same wireless domain.

INDEX TERMS Ad hoc networks, distributed scheduling protocol, exposed/hidden node, multi-channel communication, TDMA scheduling.

I. INTRODUCTION

Wireless ad hoc networks are networks without fixed infrastructure where nodes are often inexpensive devices with wireless transceivers, which could form different network topologies anywhere at any time [1], [2]. These networks operate autonomously in a self-organized manner with no central device managing the network. Typically, wireless ad hoc networks are multi-hop networks, where each node participates in forwarding data for other nodes in order to realize communication between two nodes out of direct communication range. Optionally, the nodes of an ad hoc network may be mobile. Due to low cost, easy deployment and maintenance, ad hoc wireless technology is gaining more and more attention recently with an increasing number of widespread applications being based on ad hoc technology, in particular for monitoring, surveillance, mission-critical and vehicular applications [3]. Wireless sensor networks (WSN) are a specific class of wireless ad hoc networks, forming the basis for the Internet of Things (IoT). However, despite its widespread adoption and the advantages it offers, there are still challenges that need to be solved for ad hoc technology. According to [4], there are two main challenges for ad hoc networks: Quality of Service (QoS) guarantees and scalability.

For accessing the shared wireless medium in ad hoc networks, two families of medium access control (MAC) protocols are dominant. The first family is

The associate editor coordinating the review of this manuscript and approving it for publication was Kang Chen.

contention-based protocols, typically using Carrier Sense Multiple Access (CSMA) technique. Such MAC protocols use available bandwidth on demand and are very flexible and efficient for low traffic load conditions and small network sizes [5]. When network size increases and network traffic is high, CSMA-based protocols are not able to satisfy QoS requirements, implying that CSMA-based protocols are not scalable. A second family of MAC protocols for ad hoc networks are contention-free protocols, usually based on the Time-Division Multiple Access (TDMA) mechanism. TDMA-based medium access is one of the most common medium access methods where the wireless medium is time-shared by all nodes. Channel bandwidth in the network is divided into time frames, called superframes, with every superframe further partitioned into time slots. Multi-frequency TDMA (Mf-TDMA) extends the basic TDMA medium access method, which uses only one frequency channel, to multiple channels. Slots in a Mf-TDMA superframe are represented as time-frequency tuples. In TDMA-based protocols each node transmits only during slots allocated to it, avoiding any contention for accessing the shared medium [6]. Compared to CSMA-based protocols, TDMA-based protocols mitigate internal collisions and thus improve delivered QoS for large-scale networks with high traffic demands. Due to its favorable properties in terms of scalability, TDMA scheduling techniques have gained attention for larger ad hoc networks in recent years [7]. However, the reliability and throughput of networks with TDMA access schemes may still be impacted by external interference or the occurrence of exposed/hidden nodes.

For the function of slot allocation in TDMA schemes, there are static and dynamic algorithms. As ad hoc networks need to support constant changes in traffic demands and network topology, dynamic scheduling algorithms are known to outperform static scheduling algorithms [8]. There exist two main models for handling dynamic TDMA scheduling: centralized and distributed. Centralized models consist of one or more control nodes that gather information about the network state and make scheduling decisions that are advertised to each node. In distributed models, decision-making is done at the node level based on local information on the network without requiring any centralized control; nodes exchange information about slot usage with their neighbors in order to take distributed decisions on slot allocation.

Even though centralized scheduling protocols can offer close to optimal solutions for some use cases as they have global knowledge of network topology and traffic patterns [9], they are not suitable for networks with frequently changing topology and traffic demands over time. Changes in network topology or traffic patterns result in continuous schedule recalculations and increased control overhead, thus leading to degraded network performance. Moreover, centralized scheduling protocols are not scalable as they incur high control overhead for large-scale wireless networks. In dynamic and large ad hoc networks, distributed slot allocation algorithms are preferred to cope with scalability and changes in the network topology [10]. Also, distributed algorithms are more fault-tolerant, as a major problem in centralized algorithms is the existence of a single point of failure; if the central control node fails or disconnects, slot scheduling cannot be executed anymore. In any case, whereas many distributed scheduling protocols are proposed so far, an increase in size and/or density of wireless networks still induces scalability issues for existing protocols. The most common reason for the scalability issues in large-scale ad hoc networks is their multi-hop nature, which highly depends on network size and packet forwarding capabilities [11].

Wireless networks suffer from capacity bottlenecks because the amount of available spectrum is fixed, while wireless traffic demands are rapidly growing [12]. Wireless spectrum is divided into many fixed frequency bands and most of these bands are licensed for exclusive use by specific services or wireless technologies. Licensed bands are characterized by severe overprovisioning and underutilization both in time and geographically. On the other hand, increasing traffic demands need to be supported in the presence of high interference levels for unlicensed band deployments [13]. To deal with these problems, migration to dynamic spectrum sharing techniques is a necessary step, not only in unlicensed but also in licensed spectrum bands. For supporting dynamic spectrum sharing, distributed scheduling protocols are preferred over centralized protocols because of their earlier discussed favorable capabilities to cope with dynamics and scale. To avoid interference and enable the coexistence of multiple co-located wireless networks, both in licensed and unlicensed bands, scheduling protocols should offer dynamic reallocation of slots to non-interfered sections of the available radio spectrum. Moreover, in order to increase spectral efficiency, these protocols should support spatial reuse of spectrum and tackle hidden and exposed node problems.

In large multi-hop ad hoc networks, hidden and exposed node problems represent a serious issue [14]. The hidden node problem may significantly increase the number of collisions and degrade network performance. On the other hand, the exposed node problem leads to the underutilization of the available radio spectrum and also results in reduced network throughput. In [14], it is concluded that for ad hoc networks with node density greater than 4 per collision domain, which is the case for most real-life wireless networks, the number of exposed nodes is higher than the number of hidden nodes. Hence, there is no doubt that for topologies of large-scale ad hoc networks, performance degradation is more influenced by the exposed node problem. Whilst most existing distributed TDMA protocols recognize and tackle the hidden node problem, only a minority of protocols are capable to detect the presence of exposed nodes and to minimize the huge impact in multi-hop ad hoc networks. Besides, existing protocols are mainly focusing on solving the dynamic TDMA scheduling problem for a single-frequency channel. Whereas it is possible to enhance most protocols with multi-channel TDMA scheduling capabilities, the associated overhead and scalability issues may degrade protocol performance.

To overcome the drawbacks of existing TDMA scheduling solutions, this paper presents a dynamic distributed multi-channel TDMA slot management protocol called DDMC-TDMA. Its unique design makes it a generic and scalable solution suitable for large-scale high-density networks that support dynamic traffic demands and node mobility. DDMC-TDMA mitigates the exposed/hidden node problems and is inherently robust against external interference, leading to improved wireless QoS guarantees in terms of reliability, throughput, and latency. In this work, DDMC-TDMA is presented and analyzed for ad hoc networks, but DDMC-TDMA is also more generally applicable to other types of wireless networks like infrastructure-based networks or satellite networks.

DDMC-TDMA was initially designed in the context of the DARPA Spectrum Collaboration Challenge (SC2) [15]. This paper discloses for the first time the key DDMC-TDMA features and performance, as successfully validated during SC2. DDMC-TDMA was integrated as the MAC solution for the wireless system of team SCATTER [16]. The SCATTER system proved to be one of the top-performing systems in the competition, and as an integral component of the SCATTER system, DDMC-TDMA was experimentally validated throughout the 3 years of the SC2 competition. It was experimentally validated (i) how DDMC-TDMA copes with external interference, in various challenging dynamic wireless scenarios including mobility, changing traffic patterns, single-hop or multi-hop scenarios, and (ii) how DDMC-TDMA solves hidden and exposed node problems. The main scope of DARPA SC2 was to demonstrate the feasibility of dynamic spectrum sharing in the presence of other unknown wireless technologies and interferers. However, the maximum size of wireless networks in SC2 was limited to only 10 nodes. To prove that the DDMC-TDMA protocol, as developed for DARPA SC2, can scale and adapt to a wider range of network sizes and dynamic scenarios, simulations using an ns-3 model have been executed and analyzed in this paper. Generating DDMC-TDMA models for the ns-3 environment [17] was a straightforward task as the DDMC-TDMA solution for the SCATTER system has been developed with scalability in mind.

The main contributions of this paper can be summarized as follows:

1) The proposed DDMC-TDMA protocol offers distributed slot allocation and removal functionality. It is a topology-independent solution that only relies on MAC layer control messages exchanged between neighboring nodes.
2) TDMA scheduling with low control overhead is provided for both single-channel and multi-channel environments.
3) By design, DDMC-TDMA solves both hidden and exposed node problems that may occur in large multi-hop wireless networks.
4) DDMC-TDMA represents a robust solution against interference from other networks.
5) DDMC-TDMA automatically adapts its schedule to traffic, allocating and removing slots based on changing traffic demands.
6) In-depth simulations based on an ns-3 model are performed to validate the behavior of the protocol in multiple realistic wireless scenarios with various network topologies of different sizes and diverse system requirements.

For the remainder of the paper, we present in Section II the state of the art of distributed TDMA scheduling protocols applicable to ad hoc networks. Design goals and architecture of DDMC-TDMA are explained in Section III, with a detailed description of the slot allocation and removal procedures. Validation of proposed protocol, based on multiple simulations executed, is presented and analyzed in Section IV. The paper ends with a summary of the main conclusions in Section V.

II. RELATED WORK

Several distributed algorithms for dynamic TDMA scheduling have been proposed in the existing studies for different classes of ad hoc networks.

Distributed TDMA scheduling protocol called Five-Phase Reservation Protocol (FPRP) is presented in [18]. FPRP is a heuristic TDMA slot allocation protocol that can reasonably allocate spectrum resources according to the requirements [19]. The five-phase reservation process dynamically decides the winner of each data slot, where every data slot has its own reservation slot. During reservation slots, nodes contend for the associated data slots. All the one-hop and two-hop neighbors need to approve the request for slot allocation, which leads to non-conflicting slot allocations. As all the one-hop and two-hop nodes need to approve slot allocation request, control overhead increases significantly with the increase of network size and network density. Non-conflicting slot allocation solves the hidden node problem, but FPRP only supports the reuse of the same slots by the nodes beyond two hops, which results in the occurrence of the exposed node problem. Other drawbacks of FPRP are that it is only focused on broadcast scheduling and during FPRP execution it is required that network topology does not change.

In [20] distributed randomized time slot scheduling algorithm called DRAND is proposed. DRAND represents an efficient and practical distributed scheduling algorithm [10]. It is an extension of the graph coloring scheme and it is based on heuristic centralized solution RAND [21]. RAND generates efficient slot schedules but does not offer an optimal solution and it carries all the drawbacks of centralized solutions. DRAND represents distributed implementation of RAND and thus, achieves the same channel efficiency but with increased message complexity and convergence time. This solution solves the hidden node problem, but the exposed node problem is not avoided and achieved spectrum utilization is not optimal. Wireless nodes in the DRAND algorithm work in cycles, during which control messages are exchanged between neighbor nodes. The main drawback of the DRAND algorithm is the possible occurrence of unsuccessful cycles, resulting in extra control message overhead and algorithm running time. An increase in network size and network density results in increased control overhead and increased probability of failed cycles, leading to performance degradation of the DRAND algorithm. Therefore, it can be concluded that the DRAND algorithm is not scalable.

An extension of the DRAND algorithm called Localized-DRAND (L-DRAND) is proposed in [22]. L-DRAND is a distributed slot allocation algorithm that enhances DRAND characteristics by adding features for localization and exchanges position information between network nodes. As it is a position-based algorithm, it requires that every node possess localization capability. This algorithm is based on Lamport's bakery algorithm and gives slot allocation priority to nodes close to the center of the wireless network, as it is assumed that these nodes have the largest number of neighbor nodes. L-DRAND reduces the run time of the DRAND algorithm but at the expense of an increased number of control messages with increased message complexity.

E-T-DRAND [9] is an amelioration of the DRAND algorithm where an Energy-Topology (E-T) factor is taken into account. Definition of the E-T factor is based on the influence of residual energy and topology on slot allocations. In E-T-DRAND nodes require residual energy information from their neighbors, which is exchanged via control messages and stored in an appropriate table. During a slot allocation procedure, nodes with the lowest energy level and the highest degree are given priority. This algorithm is useful in the case of WSNs, where it reduces the energy consumption of nodes with low residual energy. The performance of the E-T-DRAND algorithm is better compared to the DRAND algorithm; it improves slot allocation efficiency and reduces control overhead and energy consumption. However, as noted in [19], the message complexity, slot allocation time, and energy consumption are still too high, with a significant increase in slot allocation time for large-scale networks and the possibility of non-convergence of slot allocation time and the number of rounds.

Hence, in [19] authors proposed another amelioration of DRAND called Exponential Backoff and Energy-Topology DRAND (EB-ET-DRAND). This algorithm is based on both the E-T factor and Lamport's bakery algorithm. It uses exponential backoff to adjust slot allocation priorities. These features allow the algorithm to reduce the collision probability of control messages and message complexity compared to the DRAND algorithm. However, despite these improvements, it possesses all other shortcomings of the DRAND algorithm. On top of these drawbacks, one control slot is assigned to every data slot within the proposed frame of the EB-ET-DRAND algorithm, which results in a large portion of the spectrum allocated for the exchange of control messages. The size of the proposed frame depends on network size, with EB-ET-DRAND assuming that the upper limit of network nodes is 32 and the upper limit of neighbor nodes for each node is 8, which does not provide a scalable solution for larger networks.

In [23] a novel graph coloring technique is presented, called the Color Constraint Heuristic (CCH). Based on CCH, both Centralized Slot Assignment (CSA-CCH) and Distributed Slot Assignment (DSA-CCH) algorithms are proposed. Compared to DRAND, DSA-CCH requires fewer colors or TDMA slots, fewer control message collisions are expected, but the execution time of the algorithm is longer. DSA-CCH is not scalable as it uses a spreading approach instead of parallel algorithm execution. Execution of the algorithm at the node level is sequential; it starts at the sink/gateway and propagates through the network. Thus, the convergence of the algorithm for large network deployments becomes slow.

DRAND algorithm and its ameliorations are single-channel TDMA scheduling algorithms that store and update a superframe table per node and exchange control messages between network nodes. Ameliorated DRAND algorithms solve some of the shortcomings of the basic DRAND algorithm, but high control overhead and algorithm execution times are still present, especially in the case of large-scale networks. As such, they are more suitable for wireless networks where network topology does not change for a long period of time. The DRAND algorithm, its derivatives, and the FPRP algorithm use a greedy graph coloring approach, which is inherently sequential. This leads to inefficient and slower algorithm execution compared to synchronous algorithms. Besides, these algorithms require two-hop neighbor information, which increases execution time, decelerates scheduling convergence, and increases the number of control messages. They also try to reuse slots as much as possible, but without achieving optimal spectrum utilization.

Another example of a distributed single-channel TDMA scheme that stores slot usage information per node is disclosed in [24]. The proposed algorithm referred to as DICSA, stores a list of forbidden slots per node and enables nodes to participate concurrently in slot reservation procedures. Both aspects contribute to a more efficient slot allocation procedure. However, the proposed method does not provide a solution for the exposed node problem, leading to suboptimal utilization of the available spectrum. Also, similar to DRAND and its ameliorations, DICSA requires an affirmative response from all the neighboring nodes to reserve a requested slot. Therefore, it requires a high number of control messages for the execution of a slot allocation, resulting in scalability issues in the case of large-scale multi-hop networks.

In [25], a distributed node scheduling algorithm for multi-hop wireless networks, called Local Voting, is proposed. It is shown that the proposed algorithm achieves better performance than the other distributed algorithms in terms of average delay, maximum delay, and fairness. The Local Voting algorithm consists of two functions: requesting and releasing free time slots, and load balancing. If a node has no traffic, all its time slots are released. If a node requires new slots, it examines all the slots sequentially and the first available slots are allocated. If all slots have been allocated to one-hop or two-hop neighbors of the examined node, then no new slot is allocated to the node. This conservative approach leads to the occurrence of exposed node problem. The load balancing function is invoked to keep the load balanced between the nodes. The message exchanges for requesting and releasing slots are considered equivalent to message exchanges in the DRAND algorithm. Furthermore, same as the DRAND algorithm, the Local Voting algorithm requires the exchange of two-hop neighbor information, which leads to higher control overhead than algorithms based on one-hop neighbor information dissemination.

All algorithms presented so far are single-channel TDMA scheduling algorithms. From the respective papers, it is not clear if these algorithms can be extended to cover multi-channel environments. Consequently, if such enhancement were possible, it is unknown to which performance degradation it would lead in these more complex conditions. In the IEEE 802.15.4 standard, multi-channel communication is applied with nodes being able to switch channels quickly, or more specifically, to perform channel hoping. Sixteen non-overlapping channels are used in the 2.4 GHz band and 10 channels in the 915 MHz band [26]. This helps to avoid external interference and allows multiple simultaneous transmissions on different channels in order to increase network throughput. In recent years, the IEEE 802.15.4e standard [27], an amendment of IEEE 802.15.4 standard, proposes a redesign of the MAC layer in order to provide a framework for schedule-based communication for Wireless Personal Area Networks (WPANs) in multi-channel environments. Anyway, in [27] there is no guidance on how to assign time-frequency slots to each link, only how to apply it.

The IETF 6TiSCH working group integrated IPv6-based upper protocol stack with IEEE 802.15.4e standard [28] and introduced the 6TiSCH operation sublayer (6top) [29], which implements and terminates the 6top Protocol (6P) [30] and runs one or more 6top Scheduling Functions (SFs). 6P allows a node to communicate with neighboring nodes in order to add/remove cells, whereas the SFs define the rules when to add/remove cells, monitor performance, and collect statistics. The 6TiSCH working group has introduced the Scheduling Function Zero (SF0) [31], which is later replaced by Experimental Scheduling Function (SFX) [32]. Recently, 6top is enhanced with the Minimal Scheduling Function (MSF) [33]. Proposed SFs aim to provide a minimal set of scheduling functionalities to be usable in a wide range of applications. However, provided functionalities are basic and based on restrictive assumptions. For example, MSF is optimized for applications with regular upstream traffic from the nodes to the sink, whereas downward traffic is considered as sporadic and its management is not specified. We further differ from 6TiSCH technology by employing multi-hop Mf-TDMA scheduling without random channel hopping. Any scheduling algorithm, different from those proposed by the 6TiSCH working group, can be implemented in the 6top sublayer. A detailed literature overview of scheduling algorithms in IEEE 802.15.4e can be found in [34].

Distributed multi-hop scheduling algorithm GALLOP [35] is one example of distributed multi-hop scheduling algorithm that can be integrated with IEEE 802.15.4e standard. It has been designed to address the challenges of multi-hop closed-loop control in wireless networks with a tree topology. GALLOP supports bi-directional scheduling for cyclic information exchange and it is based on low-overhead signaling that leads to scalable operation. GALLOP also supports frequency/channel hopping for mitigating external interference. Execution of algorithm is sequential; uplink scheduling is going through ranks in tree topology from leaf nodes to sink node, whereas downlink scheduling starts from sink node and propagates through ranks of tree topology to leaf nodes. However, uplink and downlink scheduling in the GALLOP algorithm does not offer protection against exposed node problem.

A multi-channel dynamic TDMA slot reservation (DTSR) protocol is proposed in [36]. It represents an example of a protocol based on cognitive radio (CR), where it is assumed that every node is equipped with a reconfigurable transceiver and spectrum scanner. This protocol divides the spectrum into time-frequency pairs and proposes a superframe, which in addition to data and controls slots, incorporates sensing periods. During the sensing periods, nodes listen for transmissions of other nodes and acquire knowledge of free slots. DTSR requires constant traffic, as otherwise, nodes may reuse slots allocated by idle nodes, resulting in later internal collisions. Control slots are used for the exchange of control messages and every node keeps track of slot usage by its neighbors. In the case of collided control messages, DTSR does not provide a protection mechanism and it may result in superframe tables with obsolete slot usage information. Utilization of available spectrum by DTSR protocol is sub-optimal as it does not protect against exposed node problem.

DSAT-MAC [37] is a multi-channel distributed TDMA protocol where slots are dynamically distributed between CR users. This protocol allows CR users to opportunistically use unused licensed spectrum without harming the primary users that own the licensed spectrum. Same as for DTSR protocol, a prerequisite for DSAT-MAC is the spectrum-monitoring capability of every node in the network. In DSAT-MAC one control slot per node is assigned, leading to inefficient spectrum utilization, especially for large-scale networks. To acknowledge its existence, every node needs to transmit at least one control message during every single superframe, leading to an increase in control overhead. Furthermore, the execution time of protocol is high and multiple specific wireless problems may occur as hidden, exposed, and deaf node problems. Deaf node problem occurs when the transmitter and receiver are tuned on different frequencies [40].

Self-Organizing Medium Access Control for Sensor Networks (SMACS) is another multi-channel distributed protocol. It is a hybrid TDMA protocol proposed for WSNs [38]. The main advantage of this protocol is that it does not require

TABLE 1. Qualitative comparison of different scheduling algorithms/protocols.

Algorithm/protocol	Multi-channel scheduling	Hidden node avoidance	Exposed node avoidance	Supported wireless networks/topologies	Evaluation method	Sequential or synchronous execution
FPRP [18]	No	Yes	No	Static topologies	Simulation	Sequential
DRAND family [9], [19], [20], [22], [23]	No	Yes	No	Network topologies that do not change for a long period of time	Analytical [9], [20] Simulation [all] Experimental [20]	Sequential
DICSA [24]	No	Yes	No	WSNs	Simulation	Synchronous
Local Voting [25]	No	Yes	No	Multi-hop networks with static nodes	Analytical Simulation	Sequential
SFs [31]–[33]	Yes	Yes	No	Multi-hop networks	None	Synchronous
GALLOP [35]	Yes	Yes	No	Tree topology	Analytical Simulation Experimental	Sequential
DTSR [36]	Yes	Yes	No	Ad hoc networks with CR nodes	Simulation	Synchronous
DSAT-MAC [37]	Yes	No	No	Ad hoc networks with CR nodes	Simulation	Synchronous
SMACS [38]	Yes	Yes	No	WSNs	Simulation	Synchronous
STDMA [39]	No	Yes	Yes	Known topologies with a fixed number of static nodes	Analytical Simulation	Sequential
DDMC-TDMA	Yes	Yes	Yes	Topology-agnostic	Simulation[1]	Synchronous

synchronization of nodes to global time. However, the proposed protocol requires many frequency channels in the available radio spectrum, assigning randomly different frequency channels to different nodes. If there are only a few radio frequency bands and multiple nodes are assigned to operate on the same frequency bands, the possibility of overlapping slots and internal collisions is high. On another hand, due to lack of synchronization, SMACS offers non-optimal spectrum utilization with unused spectrum left between allocated slots.

Whilst the majority of distributed TDMA scheduling algorithms solve the hidden node problem, the exposed node problem and spatial spectrum reuse are not the focus of these algorithms, resulting in suboptimal spectrum utilization. Spatial TDMA (STDMA) presented in [39] is one algorithm that takes into account the exposed node problem and the fact that network nodes are usually spread out geographically. In case of sufficient spatial separation, STDMA enables nodes to reuse the same slots within the TDMA schedule. In this way, both the capacity and spectrum utilization of multi-hop networks are increased. However, STDMA is a single-channel TDMA scheduling algorithm for networks with known topologies and with a fixed number of static nodes, thus not offering flexibility to cope with the network dynamics as observed in many network deployments.

Based on the literature analysis, all existing distributed TDMA algorithms possess certain shortcomings, mainly focusing on solving a limited set of problems arising in ad hoc networks and generally disregarding scalability and spectrum efficiency. For this reason, we propose a new protocol targeting multiple dynamic use cases and solving most of the known problems occurring in distributed TDMA scheduling for ad hoc networks. Key differences between the existing state of the art and proposed algorithm are shown in Table 1. As it can be seen, our proposed solution is the only one that is topology-agnostic, while being highly scalable, solving exposed/hidden node problems, and supporting multi-channel scheduling. It is worth noting that execution times of algorithms could not be compared directly, therefore in the last column of Table 1 we present if algorithms are sequential or synchronous. In general, sequential algorithms are converging more slowly to the steady state than synchronous algorithms.

III. DESIGN GOALS AND ARCHITECTURE OF DDMC-TDMA

First, the main design goals of the proposed DDMC-TDMA scheduling approach are presented. Afterward, DDMC-TDMA architecture and its main features and procedures are described in more detail.

During the design and development of DDMC-TDMA, besides satisfying requirements for SC2, the main goal was to provide a generic solution for TDMA scheduling that would maximize spectrum efficiency while minimizing the control overhead. As centralized solutions are not scalable and cannot cope with dynamic network and traffic conditions, it was decided to go for a distributed scheduling solution. By opting for a distributed solution, single point-of-failures like in centralized solutions can be avoided. Furthermore, in a distributed solution control messages are only exchanged between neighboring nodes, limiting control overhead for spectrum management. This leads to improved performance in terms of overall system capacity and individual node throughput and latency. It was decided to further reduce control overhead by minimizing the number of control messages exchanged during a slot allocation procedure and by limiting the number of nodes participating in the procedure to only the two nodes that want to establish a link.

In addition to a slot allocation procedure, also a dynamic slot removal procedure is required to maximize coexistence with other unknown wireless networks (such as competing networks in the DARPA SC2 competition). We hereby assume that future wireless networks do not have exclusive

[1] Previously experimentally evaluated in DARPA SC2 competition [15]

access to the spectrum, but have to share the spectrum with other unknown networks. To mitigate interference, removal and re-allocation of affected slots may be needed. Moreover, traffic demands are often varying over time and it is not spectrum efficient to keep all allocated slots when traffic demands drop.

As existing distributed solutions are mostly focused on specific use cases with limited applicability elsewhere, another goal was to create a generic solution that covers a wide range of application scenarios and network topologies. The slot management protocol should support both infrastructure-based and infrastructure-less wireless networks, static as well as mobile nodes. To maximize the applicability of the protocol, it should support both single-hop and multi-hop networks operating either in a single-frequency band or multi-channel environment. The proposed MAC should also be Physical (PHY) layer agnostic, meaning that it can be transparently integrated on top of any existing or future wireless PHY standard. Another design goal was to design a protocol that maximizes throughput and reliability performance by solving the typical hidden and exposed node problems that may occur in various wireless network topologies. In other words, the protocol should guarantee full spectrum utilization with no unused slots when traffic load is high and with no overlapping slots allocated.

A dynamic distributed multi-channel TDMA (DDMC-TDMA) slot management protocol has been developed in line with the above-mentioned design goals. In this paper, DDMC-TDMA is presented and described as a scheduling solution for ad hoc networks. However, as already pointed out, the proposed protocol is also applicable for infrastructure-based networks without any loss of generality. DDMC-TDMA can be applied to both single-channel TDMA schemes and Frequency-Division Multiple Access (FDMA) schemes. If not indicated otherwise, in the remainder of this paper DDMC-TDMA is referring to Mf-TDMA access schemes, as Mf-TDMA represents the most complex case of TDMA-based spectrum sharing.

The main focus of the paper is on the MAC layer and more specifically on the TDMA slot management protocol, its applicability and scalability in different use cases. For this reason and since DDMC-TDMA has already been evaluated experimentally during SC2 competition with realistic PHY and wireless medium [16], for this work we decided to adopt a simplified PHY with symmetric links between two nodes. We further do not take into account propagation delay and transmission error models.

For TDMA access schemes to be feasible, all nodes' clocks need to be synchronized. Based on observations from the real-time SCATTER system developed for SC2, it is determined that submillisecond (sub-ms) accuracy is sufficient and this can easily be maintained with traditional synchronization methods. Therefore, it is assumed that node clocks are synchronized with this accuracy. Guard spaces are introduced to TDMA slots, which guarantees stable operation of the system as long as the clock drifts between the nodes are below 1 ms. Every node of the ad hoc network has one or more transceivers, which may work in out-of-band full-duplex or half-duplex mode. Thus, DDMC-TDMA supports distributed scheduling for networks consisting of single-radio and/or multi-radio devices.

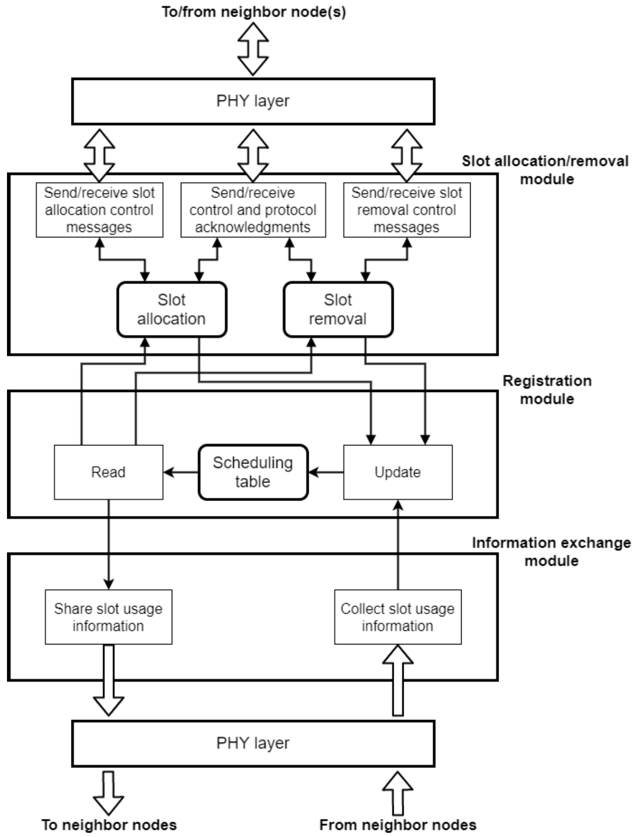

FIGURE 1. Local DDMC-TDMA entity running on each node for a distributed slot management.

The distributed slot management protocol is executed on all nodes of the ad hoc network. For local execution of DDMC-TDMA, a node does not need any knowledge regarding its position in network topology or the network size. Dynamic slot allocation/removal procedures of a node and access of a node to the shared medium are managed by an entity that is schematically presented in Fig. 1. Every node of the ad hoc network is running its own instance of DDMC-TDMA entity, allowing the node to establish communication with other nodes of the network, and therefore managing spectrum utilization in a distributed way.

The DDMC-TDMA entity for distributed slot allocation/removal consists of three modules:

1) A registration module, maintaining a scheduling table of the slots, that represents time slots and/or frequency channels.
2) An information exchange module, (i) collecting slot usage information indicative for the use of slots by neighbor nodes, (ii) updating the scheduling table based on the received slot usage information, and

(iii) sharing its own local slot usage information with neighbor nodes.

3) A slot allocation/removal module, allocating or removing a slot used for communication with a neighbor node. During the allocation/removal procedures, it consults the scheduling table and exchanges control messages with the neighbor node. In addition, it updates the scheduling table based on the outcome of the executed slot allocation/removal procedures.

In the following subsections, the modules of the DDMC-TDMA entity are described in more detail.

A. REGISTRATION MODULE

The registration module consists of a scheduling table and primitives for accessing the scheduling table. Other modules of the DDMC-TDMA entity can retrieve information from the scheduling table using the 'Read' primitive or can change the content of the table using the 'Update' primitive. The registration module is responsible for storing the most recent information of spectrum utilization as seen from a single node perspective.

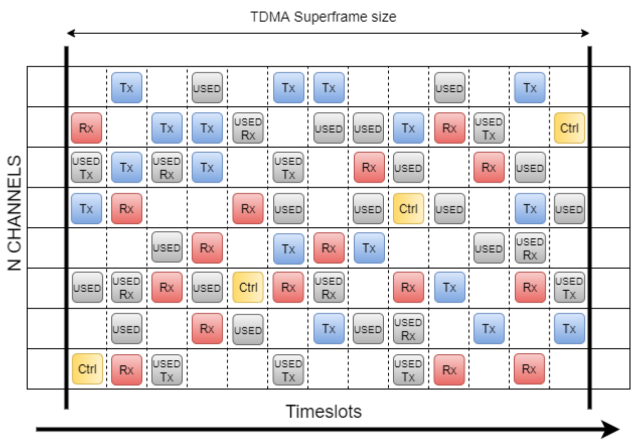

FIGURE 2. Example of a scheduling table stored per node.

1) SCHEDULING TABLE

The scheduling table can be one-dimensional or two-dimensional. For single-channel TDMA schemes, the scheduling table is one-dimensional and only consists of time slots in a single channel. An FDMA scheme is another example of a one-dimensional table, where the radio spectrum is divided into multiple channels and no time division is applied. If both time and frequency division is enabled, the scheduling table is two-dimensional and represents a Mf-TDMA medium access scheme. An example of a Mf-TDMA scheduling table kept per node is presented in Fig. 2.

Information stored in the scheduling table specifies how the slots are allocated for communication within the ad hoc network. It keeps the information about slots assigned for its own communication needs either for transmission (blue slots) or reception (red slots), and slots assigned for communication of neighbor nodes (grey slots). Some slots (yellow slots) are reserved for exchanging control messages. Only one-hop slot usage information is needed for maintaining the table, as is explained further in this paper. Scheduling tables are kept locally for each node, hence each node has its local view of the network, and scheduling tables in the same ad hoc network differ from node to node. The scheduling table is updated with every successful slot allocation/removal procedure of the node itself or any node in its vicinity. It is also updated by collecting and parsing periodic *slot usage* messages from neighbor nodes. The allocation/removal module and the information exchange module inform the registration module about any recent changes of slot usage via the 'Update' primitive of the registration module. During slot allocation/removal procedures, the slot allocation/removal module consults the scheduling table for the usage information of time-frequency slots using the 'Read' primitive of the registration module. The same primitive is used by the information exchange module for periodic reporting of the scheduling table status to neighbor nodes.

2) TYPES OF TIME-FREQUENCY SLOTS

As shown in Fig. 2, every time-frequency slot is assigned an appropriate slot type, based on the utilization of slots by the node itself or neighbor nodes. Empty cells in the scheduling table represent unused time-frequency slots, in the remainder of the paper referred to as *Empty* slots. These slots represent time-frequency slots that are not allocated by the node itself or by any node in its vicinity. Thus, these slots are free for future use in case that a new slot needs to be allocated for communication with any of the neighbor nodes.

Used slots are divided into two classes, internally and externally used slots. Internally used slots are slots allocated for communication links belonging to the specific node with its neighbors. Three types of internally used slots exist:

1) Transmission (*Tx*) slots
2) Reception (*Rx*) slots
3) Control (*Ctrl*) slots

The scheduling table also stores information about slot usage from one-hop neighbors as externally used slots. Information about usage of slots by neighbor nodes is either received during execution of distributed slot management procedures or by collecting periodical messages containing slot usage information and then is stored in the scheduling table as one of three following slot types:

1) *USED Tx* slots
2) *USED Rx* slots
3) *USED* slots

For a more comprehensive explanation of different slot types stored within scheduling tables, let us analyze Fig. 3. The left section of Fig. 3 represents an arbitrary wireless network topology with initially established communication links between the nodes. The superframe state is presented in the upper right section of Fig. 3. The superframe comprises of

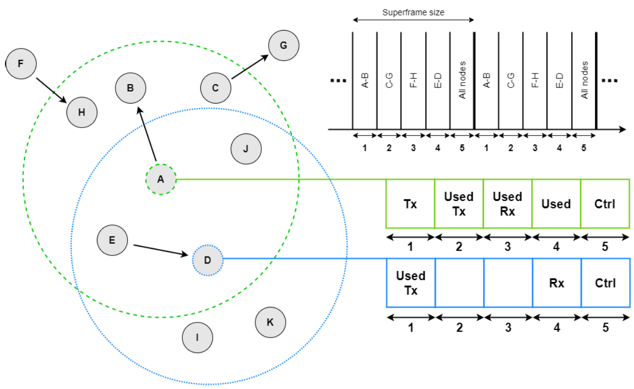

FIGURE 3. Arbitrary network topology, superframe state and scheduling tables of the network nodes.

5 time slots 1, 2, 3, 4, and 5 in a single-channel TDMA scheme with corresponding transmitter-receiver pairs in slots 1-4, corresponding to the four established links. Time slot 5 is used by all the nodes of the network for control messages. Nodes A and D and their scheduling tables are selected as explanatory examples of previously defined slot types. Therefore, collision domains of nodes A and D are illustrated in the network topology.

First of all, from the perspective of node D, time slots 2 and 3 are Empty, as all the nodes communicating within these slots are not neighbors of node D. Tx slots are slots the node uses for transmission to a neighboring node, whereas Rx slots are allocated for the reception of transmissions from a neighboring node. In the scheduling table of node A, time slot 1 is marked as Tx since node A is using this slot itself for transmission to node B. Node D is using time slot 4 itself for reception from node E, so this slot is marked as Rx in its scheduling table. Slots, marked in the scheduling table as Ctrl, are reserved for the exchange of control messages during distributed slot management protocol procedures. Time slot 5 is a common Ctrl slot in scheduling tables of nodes A and D, as well as all other nodes in the network.

Slots marked as USED Tx are allocated by a neighbor node for transmission, but none of the neighbor nodes is using them for reception. The receiver node in this case is out of range of the node in consideration. There are two slots stored as USED Tx in the scheduling tables of nodes A and D. First is time slot 2 in the scheduling table of node A, as this time slot is used for the transmission from C to G, where only node C is a neighbor of node A. As node G is not a neighbor of node A, none of the slot usage information collected by node A from nodes in its vicinity will indicate this slot being used for reception. The second slot marked as USED Tx is time slot 1 in the scheduling table maintained by the registration module of node D. Node D is only aware of the transmitter node A utilizing the slot, whereas the receiver node B is not in the vicinity of node D. USED Rx slots represent slots that are allocated by a neighbor node for reception, but the transmitter is not a neighbor of the node in consideration. Node A is storing time slot 3 as USED Rx because this time slot is used for the transmission from F to H, where the receiver node is a neighbor of node A, whereas the transmitter node is outside the range of node A. Therefore, node A cannot receive slot usage messages transmitted in a control slot by node F.

If neighboring nodes are using a time-frequency slot for both transmission and reception, that slot is labeled as USED slot. The simplest example of USED slot is a slot in which two neighbor nodes have established a communication link. This example we can see in the scheduling table of node A. Time slot 4 is marked as USED because this time slot is allocated for transmission from node E to node D and both nodes E and D are neighbors of node A. A slot is also considered as USED if one neighbor node transmits to a node out of range of node in consideration and another neighbor node is receiving at the same slot from the node out of range of node in consideration. USED Tx and USED Rx slots may transform to USED slot if subsequently reported by any neighbor node as being used for reception and transmission, respectively.

B. SLOT ALLOCATION/REMOVAL MODULE

The slot allocation/removal module supports distributed slot management on a node level with dynamic establishment and removal of communication links between any of the neighboring nodes in the ad hoc network. This module consists of two main units, the slot allocation unit and the slot removal unit.

The slot allocation unit allocates a time-frequency slot for communication with a neighbor node by executing a slot allocation procedure. The slot removal unit releases a slot allocated for communication with the neighboring node by executing a slot removal procedure. Both procedures are based on the exchange of specific control messages with the neighboring nodes. The detailed procedure for allocating or removing slots is explained in the following subsections.

Slot allocation/removal operations are event-based, reacting to a difference in the required capacity of the link compared to existing link capacity and reacting to detected interference. A slot allocation operation is triggered if the required capacity is higher than the existing capacity and it results in an increase in allocated bandwidth if there is available spectrum left. A slot removal operation is triggered if the required capacity is lower than the existing capacity and it results in a decrease in allocated bandwidth. In both cases, any ongoing communication is unaffected, with enough spectrum capacity provided by the scheduling algorithm to satisfy application demands. In case of detected interference, allocated bandwidth stays the same, with interfered slots reallocated to non-interfered parts of the spectrum. As an inevitable consequence of interference impact, application flows utilizing interfered slots may experience a drop in throughput, until reallocation of slots is executed. Slot allocation/removal operations do not have any negative impact on application flows themselves, with only increasing or decreasing utilized bandwidth in order to incorporate all application traffic while minimizing spectrum footprint.

1) SLOT ALLOCATION PROCEDURE

The slot allocation unit allocates a slot for communication with a neighboring node. During a slot allocation procedure, both the initiating node and neighbor node are involved. To agree on a slot to be allocated, their slot allocation units exchange slot allocation control messages over control slots. The procedure is initiated by the transmitter node upon demand for increasing the application throughput. The slot allocation unit of this node consults its scheduling table using the 'Read' primitive of its registration module in order to select available slots that can be proposed for establishing a communication link with the receiving neighbor node. The receiver node's slot allocation unit consults its respective scheduling table to determine which slot, from the list of slots proposed by the transmitter, may be selected for reception. Once both nodes agree on a slot, the communication link is established and the scheduling tables are updated using the 'Update' primitive of registration modules. The type of the allocated slot is Tx at the transmitter side and Rx at the receiver side. In this way, each node keeps track of the slots that have been allocated for active communication links with neighboring nodes.

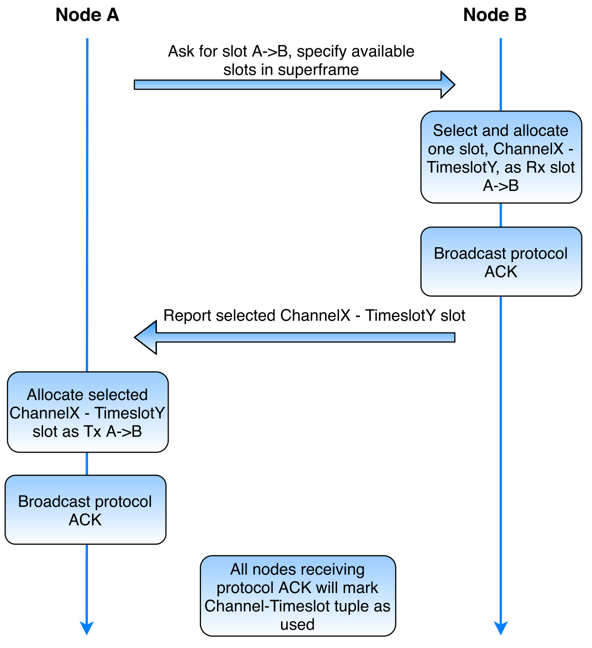

FIGURE 4. Flow diagram of a slot allocation procedure.

Fig. 4 illustrates the flow diagram of a slot allocation procedure and its control messages that are exchanged between the medium access entities of two neighbor nodes A and B.

If node A has a frame(s) to send to node B and there are not enough allocated slots for transmission to B, node A should initiate a slot allocation procedure in order to satisfy application data rate demands. To get an overview of the slots available to be allocated as transmission slots, node A consults its scheduling table. A list of available slots are filtered based on the following rules:

- Empty slots in the scheduling table are not used by the node itself or any of the neighbor nodes and can all be proposed for a new wireless connection.
- USED Tx slots are allocated by one neighbor node of node A for transmission, but no other neighbor node of node A is using this slot for reception. This implies that the receiver for this particular communication link is outside the neighborhood of node A and will not be affected by transmissions of node A. For this reason, such a slot may be a part of the proposed slots list.
- All other slot types cannot be proposed.

Next, node A selects one or more slots from the list of available slots. These slots are proposed to node B through a *proposed slots* control message. The maximum number of proposed slots that can be embedded in the proposed slots message depends on the transport block size supported by the PHY layer.

Upon reception of the proposed slots message, node B consults its scheduling table and selects one slot, further referred to as the allocated slot, from the list of the proposed slots. For the slot selection, node B applies the following rules:

- Empty slots are not used by node B or any of the neighbor nodes and are therefore available for selection.
- USED Rx slots are allocated by one neighbor node of node B for reception, but no other neighbor node of node B has allocated it for transmission. This implies that the transmitter for this particular communication link is outside the neighborhood of node B. Therefore, a USED Rx slot is available for reception by node B, as there is no risk for interference by other transmitters using the same slot.
- Other slot types are considered as unavailable for selection by the receiver node.

After selecting one slot from the filtered list, node B updates its scheduling table, so that the corresponding time-frequency tuple is marked as Rx. Node B reports the allocated slot by transmitting a *selected slot* control message to node A. Upon reception of the selected slot message, node A updates its scheduling table accordingly by marking the newly allocated slot as Tx. At the end of a slot allocation procedure, both nodes A and B have allocated the same slot and their scheduling tables are updated and aligned. As a result, they have successfully set up a new communication link or have increased the capacity of an already existing link.

The final step of a slot allocation procedure is to notify all neighbor nodes about the allocated time-frequency slot. Neighbor nodes are notified by control messages called *protocol acknowledgment* (ACK), broadcasted by the two nodes participating in the slot allocation. Protocol ACK messages are carrying information related to the outcome of the executed slot allocation procedure and upon reception of such a message, other neighbor nodes can update their scheduling tables accordingly.

By following the described steps for proposal and selection of adequate slots, DDMC-TDMA inherently avoids the hidden node problem. On the other hand, storing the USED Tx and USED Rx slot types in the scheduling table and distinguishing them from the USED slot type, prevents these types of slots from being unduly considered as unavailable for allocation. Exploiting these slot types during a slot allocation procedure enables DDMC-TDMA to solve the exposed node problem.

Other general rules may apply for slot selection narrowing down the number of slots that may be selected, such as:

- The number of simultaneous transmit-receive pairs using the same time slot is limited by the number of transceivers supported by the node and the capability to support the full-duplex operation.
- If only one transceiver is supported by the node, the same time slot of a Ctrl slot cannot be allocated for other transmit-receive pairs.

If there are multiple appropriate slots for slot allocation, different strategies can be applied for slot selection:

- The most simple strategy is when the receiver node randomly selects one slot from the list of proposed slots.
- The receiver node can select a slot based on its position in the scheduling table, for example by first filling unused frequencies within a certain time slot.
- The receiver node can select a slot based on a weight assigned to the slot. An example of such a weight parameter could be the presence of external interference measured by spectrum monitoring.
- In order to optimize spectrum utilization, the strategy can be driven by prioritizing already used slots, marked as USED Tx and USED Rx in scheduling tables. This strategy is highly recommended when multiple wireless networks have to share the same spectrum and it is further applied in this paper.

2) SLOT REMOVAL PROCEDURE

By releasing allocated slots that are underperforming or unused, medium access control can dynamically respond to changes in application layer demands or changes in the network that affect the reliability of established links. Both transmitter and receiver node participating in a link can initiate a slot removal procedure, where an established communication link may consist of one or more time-frequency slots. The reliability and efficiency of every allocated slot are monitored over consecutive superframes. If the performance of a specific slot is not satisfying the application requirements for a predefined number of superframes, then this unreliable slot needs to be released, i.e. making it available again for another allocation by other nodes. Another trigger for initiating a slot removal procedure is when a slot is idle for a while (idle period) due to decreased application demands or because an application has ended.

The receiver node monitors the presence of network traffic within allocated Rx slots, whereas the transmitter node monitors the quality in terms of successfully delivered frames in its allocated Tx slots. If the Packet Error Rate (PER) value for any Tx slot is above a predefined threshold (PER removal initiation threshold) for a predefined number of consecutive superframes (poor quality period), the transmitter node initiates a slot removal procedure for that slot. A high PER could be caused by interference from an unknown external network. Slots might also suffer from internal interference, which can be caused by collisions between control messages or by the distributed nature of the slot management protocol. By removing unreliable slots, both external or internal collisions may be resolved.

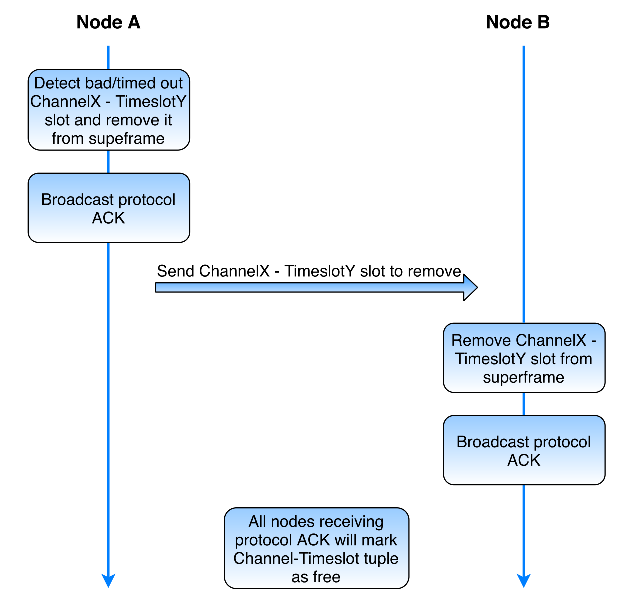

FIGURE 5. Flow diagram of a slot removal procedure.

Fig. 5 illustrates the flow diagram of a slot removal procedure executed between two neighbor nodes A and B that have established a link for communication. The flow diagram of a slot removal procedure is the same regardless of the initiator being the transmitter or the receiver. Slot removal units of nodes A and B will remove the previously allocated slot for communication with a neighbor node by consulting their scheduling tables, detecting unreliable or idle slots, and exchanging slot removal control messages.

After node A consults its scheduling table and detects a slot that is unreliable or idle for a predefined number of consecutive superframes, it releases the slot and notifies node B participating in communication over the detected slot. By utilizing shared control slots, node A transmits *remove slot* message to node B, advertising the time-frequency slot to be removed. Upon reception of remove slot message, node B also releases the underperforming slot. After the successful release of the slot, the corresponding slot type Rx information in the scheduling table of the receiving node is deleted.

Similarly, the slot type Tx information in the scheduling table of the transmitting node is deleted. As such, after the execution of the removal procedure, the previously allocated slot is transformed into an Empty slot in the scheduling tables of nodes A and B. Both nodes A and B broadcast protocol ACK message with information about the slot removed, allowing all neighbor nodes of nodes A and B that can overhear broadcasted protocol ACKs to update their scheduling tables accordingly.

C. AUXILIARY CONTROL MESSAGES IN SLOT ALLOCATION/REMOVAL PROCEDURES

In addition to primary control messages for slot allocation/removal procedures, slot allocation and slot removal units of MAC entity for distributed slot management use two auxiliary types of control messages. Those control messages are *control ACKs* and protocol ACKs. Control ACKs are exchanged between the nodes participating in a slot allocation/removal procedure to detect if primary control messages are interfered and should be retransmitted. As previously explained, protocol ACKs are used to notify neighboring nodes about the outcome of executed procedures.

1) CONTROL ACKs

Control ACKs have the same purpose for control messages as data ACKs have for data messages. Usage of control ACKs increases the reliability of the control messages exchanged during the execution of the protocol's procedures. Just like other control messages, control ACKs are transmitted in one of the reserved control slots. Every successfully received unicast control message is followed by the transmission of a control ACK message from the involved receiver node. If a unicast control message is not successfully acknowledged, then the transmitter will retransmit the control message, increase its retransmission counter *Nretr* and continue monitoring for a corresponding control ACK. The same procedure is repeated until a control ACK message is successfully received or the maximum number of control message retransmissions *Nretr_max* is reached, as illustrated in Fig. 6. The value for *Nretr_max* is predefined and may be different for different types of control messages. The introduction of control ACKs in the distributed slot management protocol increases the load in the control slots, but at the same time increases the reliability of control message transmissions and the success rate of slot allocation/removal procedures.

2) PROTOCOL ACKs

Information about time-frequency resource usage of neighbor nodes is collected in two ways. The first way is based on the control messages exchanged by neighbor nodes during slot allocation/removal procedures and is event-based. Each time a slot allocation/removal is executed, each involved node broadcasts one special protocol ACK message, to inform other nodes in their communication range about the outcome of the procedure. A broadcasted protocol ACK consists of the time-frequency tuple and type of slot. In the case of slot

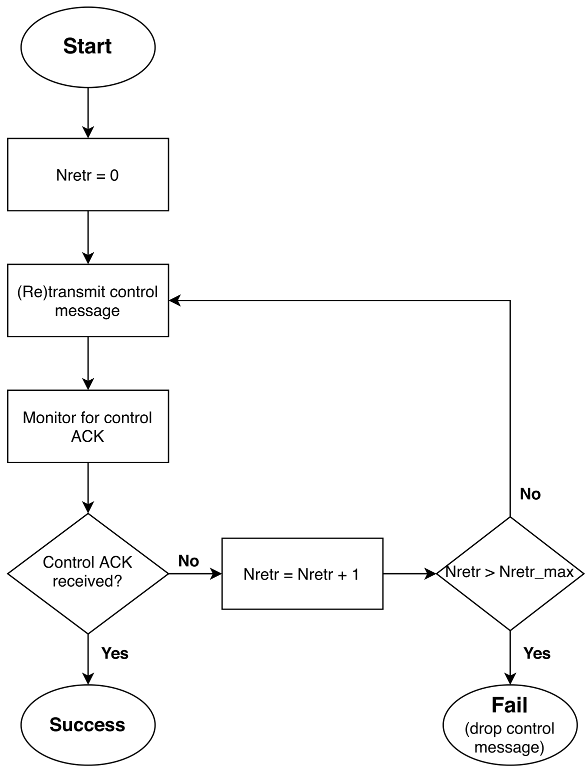

FIGURE 6. Flow diagram of a control message transmission procedure.

allocation procedure, the protocol ACK includes the type of allocated slot, either Tx or Rx type of slot. For slot removal procedures, the protocol ACK includes information on the type of slot that is removed, either Tx or Rx type of slot.

By using protocol ACKs in the slot allocation/removal algorithm, neighboring nodes in the network are immediately notified about the most recent changes in slot usage allowing them to update their scheduling tables with the latest usage information, which has a positive impact on the reliability and effectiveness of the slot management protocol. However, in ad hoc networks with mobile nodes, protocol ACKs may not be sufficient to cope with mobile nodes that were initially out of range and later enter the communication range. It may also happen that protocol ACK messages are lost due to interference, preventing the update of time-frequency resource usage information in the scheduling tables. Therefore, a second way to inform neighbor nodes about time-frequency resource usage is introduced. This approach is based on the information exchange modules periodically broadcasting scheduling tables of the network nodes, which is explained in more detail in the following subsection.

D. INFORMATION EXCHANGE MODULE

The information exchange module is responsible for periodically sharing information on scheduling tables between nodes and is supplementary to the slot allocation/removal algorithm in case protocol ACKs are missed. Scheduling tables are embedded into slot usage messages exchanged between the information exchange modules of the MAC entities in order to support distributed slot management for more dynamic situations where the slot allocation/removal algorithm does not suffice. Based on the slot usage information collected by the 'Collect slot usage information' unit of the information exchange module, the scheduling table is updated. The information exchange module further comprises a sharing unit, sharing the content of the scheduling table with neighbor nodes. The 'Share slot usage information' unit accesses the scheduling table and retrieves information about time-frequency slots used by a node itself, either for transmission or for reception. It further embeds this information into a slot usage message and broadcasts it. This allows neighbor nodes to acquire knowledge about the slot usage of nodes in the one-hop neighborhood. If information about slot usage carried by a slot usage message differs from the status in the scheduling tables or if usage information is missing, the receiving nodes update their scheduling tables accordingly.

Slot usage messages are being broadcasted periodically, with a calculated period of scheduling table broadcasting (Tsh). In order to avoid multiple broadcasts of slot usage messages by different nodes simultaneously and to avoid the congestion of control slots, Tsh consists of a fixed scheduling table broadcasting delay ($Tshf$) and a random scheduling table broadcasting delay ($Tshr$) as given in (1).

$$Tsh = Tshf + Tshr \tag{1}$$

The fixed part introduces a minimum period between two consecutive slot usage messages transmitted by a node, whereas the random part introduces randomness in the time interval between slot usage transmissions.

Periodic broadcasting of slot usage messages ensures that all nodes in the network acquire an up-to-date view on occupied slots and slots available for allocation, in particular in dynamic network topologies with mobile nodes. Slot usage information collected from periodic broadcast reports complements slot usage information retrieved during event-based slot allocation/removal procedures. Such a combined approach leads to a more reliable slot management protocol that converges faster to a steady state.

E. CONTROL SLOTS

In DDMC-TDMA, a predefined number of control slots, marked as Ctrl in the scheduling tables, are reserved for exchanging control messages. These slots have the same time-frequency positions for every node of the ad hoc network. Control slots are used for broadcasting periodic slot usage information and for the execution of slot allocation/removal procedures. During the execution of slot allocation/removal procedures, both primary and auxiliary control messages are exchanged via control slots.

1) ALLOCATION SCHEME OF CONTROL SLOTS

In DDMC-TDMA, initial control slots are allocated following a static scheme during boot time. The number of control slots depends on the total number of time-frequency slots present in the available radio spectrum; it depends on the channelization of the spectrum bandwidth and on the number of time slots in the superframe. The more time-frequency slots in the superframe, the more control slots are needed to offer sufficient opportunities for slot allocation/ removal procedures to maintain reliable operation and to achieve fast convergence of the slot management protocol. However, a higher number of control slots also reduces the time-frequency resources available for the allocation of data slots, thus reducing the overall capacity of the network. This implies that fine-grained superframes (with many narrow frequency channels and small time slots) will be more spectrum efficient than course-grained superframes (with a limited number of channels and large time slots), as in the former case the ratio of control slots to data slots can be lower.

A minimal number of control slots, nicely distributed over the superframe, is required to minimize the impact of interference. The control slot allocation scheme is configured for the maximal distribution of the slots across time and frequency axes. If a limited number of control slots is interfered, other non-interfered control will still allow proper execution of slot allocation/removal procedures.

All nodes in the same ad hoc network adopt the same scheme of common control slots and update their scheduling tables accordingly. This allows neighboring nodes within the same ad hoc network to exchange control messages during the same time-frequency slots. Multiple nodes can concurrently execute slot allocation/removal procedures during the same control slot limiting execution times and allowing them to react swiftly to changing and new traffic demands in the network.

2) MEDIUM ACCESS SCHEME FOR CONTROL SLOTS

To manage the access of multiple nodes to a common control slot, an appropriate medium access protocol is required. For example, random access protocols may be used like pure aloha, slotted aloha, or CSMA with Collision Avoidance (CSMA/CA). In this paper, we have adopted an enhanced slotted aloha protocol. Other medium access schemes may be more efficient, but optimizing the medium access scheme for control slots was not the focus of this paper.

Slotted aloha requires time synchronization of the nodes. To support TDMA access, it is already assumed that nodes are synchronized with sub-ms accuracy. As it is shown in [41], to achieve maximum efficiency of slotted aloha protocol, nodes are also supposed to know the number of neighbor nodes which can be different in different parts of the network and can further change when mobile nodes are present in the network. In DDMC-TDMA, nodes acquire knowledge of their neighbor nodes by exchanging and parsing periodic slot usage messages and protocol ACK messages. Therefore, the slotted aloha protocol has been enhanced with a random slot selection that dynamically adapts to the number of neighbor nodes. This protocol enhancement is also beneficial for supporting mobility. Please note that the slot size employed in the slotted aloha scheme is a fraction of the TDMA slot size, as the slotted aloha scheme is executed within control TDMA slots.

3) RELIABILITY ENHANCEMENT OF CONTROL SLOTS

To prevent control slots from being saturated and becoming unreliable, the number of nodes that simultaneously execute slot allocation/removal procedures should be limited. To cope with this problem, slot allocation procedure timeout (*Talloc*) and slot allocation procedure delay (*Twait*) are introduced. *Talloc* is a fixed value, whereas *Twait* is a random value between predefined *Twait_min* and *Twait_max* values, i.e. *Twait* is calculated as:

$$Twait = \text{rand}(Twait\_min, Twait\_max) \qquad (2)$$

where rand(a, b) is a function that returns a pseudo-random number in the range between a and b.

The concept of timeout *Talloc* and delay *Twait* is going to be explained in the example presented in Fig. 4. Upon the start of a slot allocation procedure, node A sets a timeout *Talloc* in which the procedure is expected to finish. If the procedure is executed successfully within this time frame, timeout *Talloc* is discarded and node A may initiate subsequent slot allocation/removal procedure after the *Twait* period. If the procedure is not finalized in time, this indicates a heavy load on control slots; control messages between nodes A and B are colliding with control messages from other network nodes. In this case, node A waits until timeout *Talloc* expires and consequently backs off from using control slots while they are used heavily by other nodes in the network. Besides, node A has to wait for the random delay *Twait* before starting the next slot allocation/removal procedure. Imposed randomness reduces the probability of the large number of nodes starting simultaneous slot allocation/removal procedures and increases the reliability of control message transmissions in control slots.

IV. VALIDATION AND EVALUATION

To evaluate the correct operation and performance of the proposed DDMC-TDMA protocol, sets of simulation runs are executed for various ad hoc network configurations. Firstly, the basic functionality of slot allocation/removal procedures is evaluated in the case of a single-hop wireless network, operating both isolated and in presence of unknown external interference. Furthermore, the scalability of the DDMC-TDMA protocol is verified by increasing the number of devices in single-hop and multi-hop networks, as well as the network density of multi-hop networks. Network density represents the number of nodes in a single collision domain. Besides verifying that the protocol is scalable, the size and density of single-hop or multi-hop ad hoc networks are varied to prove that protocol is also able to achieve full spectrum utilization with an increasing number of network nodes. Multi-hop network topologies are simulated to prove the efficiency of protocol in the case of large networks covering multiple collision domains, with a focus on spectrum reuse. Finally, multi-hop topologies are analyzed to validate how the proposed protocol mitigates the hidden and exposed node problems occurring in such network topologies.

The DDMC-TDMA protocol is simulated and analyzed in the ns-3 simulator environment. The implementation of the DDMC-TDMA protocol in ns-3 is available online [42]. For every use case and its subset, 20 independent ns-3 simulations were conducted. The mean value of convergence times to steady state (*Mconv*) and confidence interval for an adopted confidence level of 95% were calculated for each set of simulation runs. Simulation runs with convergence times outside of calculated confidence interval *Mconv* ± 3.5% were discarded and remaining results averaged. To avoid seed-related artifacts in simulation results, a random seed is generated from the simulation number and every node is booted at a random time at the start of the simulation. Please note that all the results presented in this section are analyzed in line with the start of the simulation and not after simulation warm-up time has passed.

TABLE 2. Default parameters used for simulations.

Simulation parameter	Value
Simulation duration	820 s
Superframe duration	1000 ms
Slot duration	50 ms
Slot's slice for data frames	43 ms
Slot's slice for ACKs	4 ms
Slot's slice for guard spaces	3 ms
Number of time slots in superframe	20
Number of data time slots in superframe	16
Number of control time slots in superframe	4
Maximum Tx/Rx slots allocated per node	16/16
Application data rates	400 and 800 packets/s
Maximum achievable frame rate	688 frames/s

Default parameter values for the executed simulations are presented in Table 2. It is worth noting that most of the simulation parameters are selected based on the implementation of the SCATTER system. For example, PHY-related parameters as transmission duration of a single frame and minimum single frame size are selected based on LTE-based SCATTER PHY layer characteristics [43]. The number of data and control slots and their durations are selected in order to satisfy QoS demands imposed during the DARPA SC2 competition.

Based on the parameters from Table 2, the Mf-TDMA superframe state for the performed simulations is presented in Fig. 7. Whereas the number of time slots is fixed, the number of channels varies and is specified later for every use case. It is assumed that all nodes in the network are comprised of a single transceiver supporting out-of-band full-duplex operations, so all the devices can simultaneously allocate Tx and Rx slots in the superframe at adjacent channels. As nodes both listen and transmit during control slots, for the duration of control slots no data transmission can be performed at adjacent channels. Four control slots are present within the superframe, and as a result, every node can allocate up to

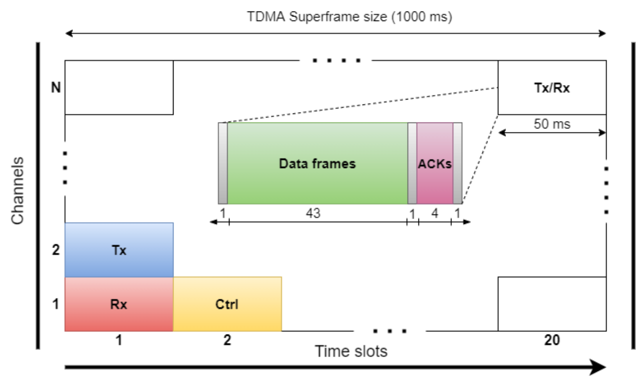

FIGURE 7. Mf-TDMA superframe state for the performed simulations.

16 Tx and 16 Rx slots. As presented in Fig. 7, data slots within the superframe are supporting fast ACKs, used for determining the PER value of slots. Slots may also be configured in any other way, as long as there is a reliable way to calculate slot quality. Within the adopted slot structure 43 ms are used for data transfer. With assumed LTE-based PHY, this leads to a maximum of 43 data frames transmitted per slot. The receiver reports the successful reception of data frames by transmitting ACKs. To conform with the minimum single frame size, every ACK carries a maximum of up to 11 confirmations of successful frame receptions. Therefore, the receiver transmits 4 ACKs in total, occupying 4 ms per slot. The remaining 3 ms within a slot are used as guard space, allowing sub-ms time drift between two nodes participating in the communication, thus simulating time drifts probable in real-time systems.

The maximum transmitting capacity of a node is calculated as the maximum number of Tx slots multiplied by the maximum number of frames supported per slot. As a result, the maximum achievable data rate of a node is 688 frames/s. For analyzing system behavior with different application loads, two application data rates are introduced, with one application packet fitting into a single MAC frame. The data rate of 800 packets/s, referred to as a high data rate, is the data rate that saturates the capacity of a transmitting node. The data rate of 400 packets/s is below the maximum supported data rate and in the remainder of the section is called low data rate.

Adopted values of parameters described in Section III for the proposed DDMC-TDMA protocol are given in Table 3. Optimal values for default parameters of DDMC-TDMA protocol were deduced by performing multiple simulations in different wireless network topologies consisting of a different number of nodes. Surely, different values would result in better performance for specific use cases, but values given in Table 3 are generic for all use cases presented in this section. It is worth noting that a high value for PER removal initiation threshold is adopted, as the system also needs to work with scenarios with high network load and interference conditions.

TABLE 3. Default parameters for DDMC-TDMA protocol.

Protocol parameter	Value
Maximum number of proposed slots	10
Idle period	5 superframes =>5000 ms
Poor quality period	2 (fixed) or rand(2,5) superframes =>2000 or 2000-5000 ms
PER removal initiation threshold	20%
Maximum number of control message retransmissions (*Nretr_max*)	3
Fixed scheduling table broadcasting delay (*Tshf*)	4000 ms
Random scheduling table broadcasting delay (*Tshr*)	rand(0,4000ms)
Period of scheduling table broadcasting (*Tsh*)	4000 ms + rand(0, 4000) ms
Slot allocation procedure timeout (*Talloc*)	12000 ms
Twait_min	2500 ms
Twait_max	3500 ms
Slot allocation procedure delay (*Twait*)	rand(2500, 3500) ms

A. USE CASE 1. SINGLE-HOP COLLISION DOMAIN NETWORKS AND SCALING EXAMINATION

In the first use case, simulations are executed to validate the basic functionality and scalability of the proposed protocol for wireless networks operating in a single-hop wireless domain. The number of nodes is gradually increased from 10 to 50 in steps of 10. Every node in the network establishes two communication links with other nodes of the network. In one of the links, the node is acting as a transmitter, whereas in the other link as a receiver. A high application data rate is assumed, thus every node tries to allocate the maximum number of transmitting slots within the superframe.

Three different sets of simulations were conducted, with the number of executed slot allocation/removal procedures and convergence time to steady state analyzed and compared. In the first and second set of simulations, the number of available frequency channels is set to 16, whereas in the third simulation, the number of channels is 50% higher than the number of nodes in the network. Therefore, in the first two sets of simulations, full spectrum utilization is expected as nodes have to compete for a limited spectrum. In the third set of simulations, two-thirds of the available spectrum is expected to be allocated for network communications. As there is enough available spectrum to satisfy network throughput requirements, every node is guaranteed enough data slots to satisfy its application demands. Another difference between the sets of simulations is in the adopted value for a poor quality period after which slot removal operations are initiated. In the first and third sets of simulations, a fixed value from Table 3 is adopted. For every allocated slot in the second set of simulations, the poor quality period is randomly calculated as an integer number between 2 and 5 consecutive superframes. In summary, the first set of simulations is conducted with basic DDMC-TDMA, the second set with DDMC-TDMA with randomized initiation of high PER removal operations, and the third set with basic DDMC-TDMA in an environment with more frequency channels available.

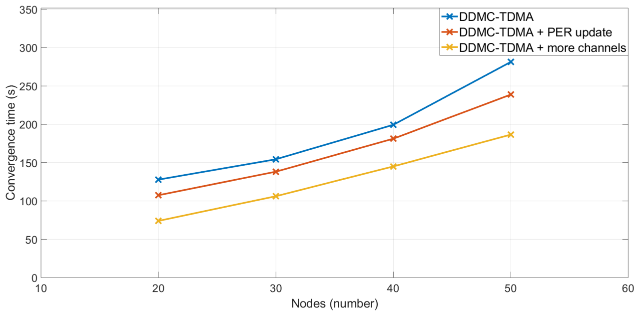

FIGURE 8. Convergence times for 3 sets of single-hop simulations.

Convergence times, being the times needed by the networks to achieve a steady state, are presented in Fig. 8 for the three sets of simulations. A number of executed slot allocation (A) and slot removal (R) procedures, averaged over 20 independent ns-3 simulations, are shown in Table 4. From Fig. 8 and Table 4 it can be concluded that for the three sets of simulations, convergence time increases with the number of nodes and the number of executed slot allocation/removal procedures. Nevertheless, a steady state with full spectrum utilization is achieved in all three sets of simulations regardless of network density.

TABLE 4. Averaged numbers of the executed slot allocation (A)/removal (R) procedures until a steady state is reached.

Number of nodes	Basic DDMC-TDMA	Basic DDMC-TDMA + random PER removal	Basic DDMC-TDMA + more channels
20	A: 304.80 R: 60.70	A: 299.80 R: 54.0	A: 332.50 R: 25.10
30	A: 405.20 R: 184.90	A: 378.40 R: 154.50	A: 503.40 R: 70.0
40	A: 565.0 R: 382.30	A: 545.0 R: 352.60	A: 681.10 R: 131.80
50	A: 739.10 R: 613.80	A: 706.0 R: 544.70	A: 851.30 R: 210.60

There are a couple of factors leading to increased convergence times with increasing network density. More communicating network nodes lead to higher congestion of available control slots. More control messages need to be exchanged and, hence, there is a higher probability of control messages colliding. As such, slot allocation/removal procedures are less reliable and multiple retransmissions of control messages might be required to complete initiated procedures, resulting in increased execution times of procedures and consequently increased convergence times. If control slots get saturated, it might even lead to failure in finalizing some of the initiated slot allocation/removal procedures, causing them to timeout. Heavy utilization of control slots might also lead to nodes missing slot usage reports, i.e. protocol ACKs or slot usage messages. Missed slot usage reports, asynchronous execution of the protocol in the different nodes of the network, and the randomness of slot selection might result in two or more nodes allocating the same slot simultaneously. As a result, communication links within overlapping slots will interfere, and a high PER removal procedure is triggered and executed by one or more nodes. The probability of overlapping slot allocations is proportional to the active communication links and inverse proportional to the available radio spectrum, which is limited for the first two sets of conducted simulations.

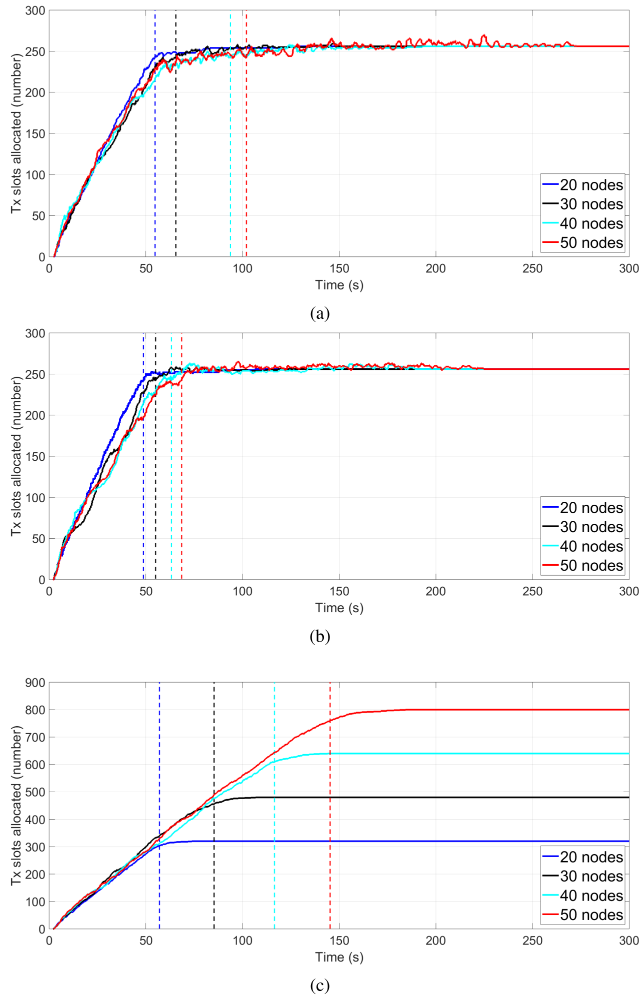

FIGURE 9. Tx slot allocation graphs. (a) Basic DDMC-TDMA. (b) Basic DDMC-TDMA + random PER removal. (c) Basic DDMC-TDMA + more channels.

In Fig. 9 Tx slot allocation graphs are presented. Vertical dashed lines represent a point in time after which the number of allocated Tx slots within the network stays above 95% of its maximum achievable value. It can be seen that for the first two sets of simulations, approximately above 95% slot allocation is achieved relatively quickly. Afterward, network nodes struggle to successfully allocate and preserve the remaining unallocated slots. All nodes in the network are trying to asynchronously allocate the last couple of slots to satisfy their throughput demands. For the last 5% of the slots, a limited number of nodes are allocating overlapping slots followed by initiating slot removal procedures. This process continues until the nodes finally allocate slots that do not overlap with slots allocated by other nodes. A prolonged period of slot allocations and removals when only a limited number of free slots (or unutilized spectrum resources) is available, influences the convergence time significantly.

With the random calculation of a poor quality period instead of a fixed poor quality period, the probability that one link remains active within a slot allocated by several links is increased, while other links vacate the slot, thus solving internal interference. This adjustment of basic DDMC-TDMA leads to reduced convergence time and the number of allocation/removal procedures for all network densities (see Fig. 8 and Table 4).

Despite using a fixed value for poor quality period threshold, the third set of simulations with more available channels offers the fastest convergence time, as seen in Fig. 8. As more radio spectrum is available, overlapping slot allocations are less probable. In Fig. 9 we can indeed see that the number of allocated Tx slots is almost linearly increasing until a steady state is achieved. The oscillatory behavior (due to prolonged allocations and removals of the last couple of slots) observed in the first two sets of simulations, does not occur in the third set of simulations. Reaching the limits of the available spectrum resources is hence the most significant factor for increasing the convergence time. This is also confirmed in Table 4, where the third set of simulations displays much fewer removal procedures compared to the first two sets of simulations. The number of allocations for this set of simulations is higher, which can be attributed to more available frequency channels and accordingly more data slots to allocate.

B. USE CASE 2. SINGLE-HOP NETWORKS WITH EXTERNAL INTERFERENCE

With increasing wireless demands, also the probability of multiple wireless networks residing in the same geographical area is increasing, leading to mutual interference and reduction in achieved QoS. To satisfy reliability and throughput requirements, networks running DDMC-TDMA can detect time-frequency slots interfered by external networks and reallocate slots to unused slots that are not interfered.

In this use case, simulations are conducted with ad hoc networks using DDMC-TDMA and operating in the presence of other unknown external networks in their collision domain. These external networks may form any topology, use any medium access method, and do not employ coexistence and collision avoidance techniques. The ability of DDMC-TDMA based networks to mitigate external interference from these unknown external networks is examined. Sizes of DDMC-TDMA based networks are ranging from 10 to 50 with an incremental step of 10. For all executed simulations, it is assumed that external networks are inactive at the beginning of the simulation and start operating in the shared radio spectrum 250 seconds after the start. There are sufficient spectral resources, meaning sufficient time-frequency slots, to serve all traffic demands in the DDMC-TDMA based network and external networks (number of channels is 50% higher than the number of nodes in the DDMC-TDMA network). Spectrum usage of external networks is varied per simulation, so that induced interference to the DDMC-TDMA network is in the range of 1-5 Tx slots per transmitter node.

The throughput graph of the ad hoc network consisting of 10 nodes and operating in presence of unknown external networks with different spectrum footprints is presented in Fig 10. It can be seen that after an initial drop in the throughput, the DDMC-TDMA network will recover to its initial status by removing the interfered slots and allocating new slots that are free from interference. Regardless of the size of interference, the network employing DDMC-TDMA can mitigate the interference and turn back to stable operation meeting throughput demands. However, as is expected, throughput drop and recovery time are higher in case of more interfered slots.

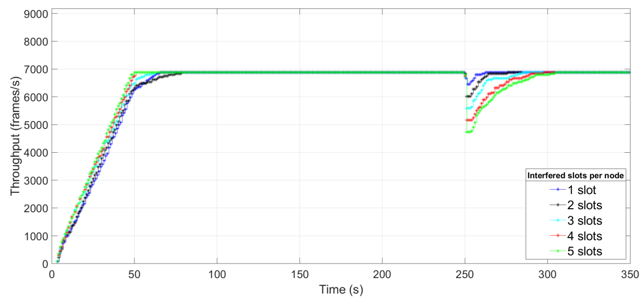

FIGURE 10. Throughput graph of 10 nodes network with different external interference.

An example of accomplished external interference mitigation and reallocation of interfered slots to interference-free slots of the radio spectrum can be seen in Fig 11. In this figure, the size of the DDMC-TDMA network is 10 nodes and the interference from external networks is 5 Tx slots per node, resulting in 50 interfered slots in total. The upper part of Fig 11 shows the steady superframe state of the ad hoc network, which is achieved before the impact of external interference, whereas the lower part of Fig. 11 shows how the interfered slots have been reallocated to slots in the non-interfered section of the radio spectrum. In this way, the DDMC-TDMA network maintains its initial performance, while the external networks keep operating in the spectrum they occupied at the 250th second.

Interference recovery times for DDMC-TDMA networks with varying sizes and with a different number of externally interfered Tx slots per node are presented in Fig. 12. Recovery times increase with the number of interfered Tx slots per node for all network sizes, and also with increasing network size. This is in line with the expectations, as with more nodes in the network, more slots are interfered, and more slots need to be reallocated.

An additional increase of recovery times may be attributed to the reallocation procedure, where due to the asynchronous nature of the protocol, multiple links may allocate overlapping slots, as previously explained in the case of single-hop networks. In the conducted simulations, external networks occupy fixed parts of the radio spectrum. Nevertheless, DDMC-TDMA is able to mitigate dynamic interference

FIGURE 11. Superframe state of 10 nodes network before and after mitigation of external interference.

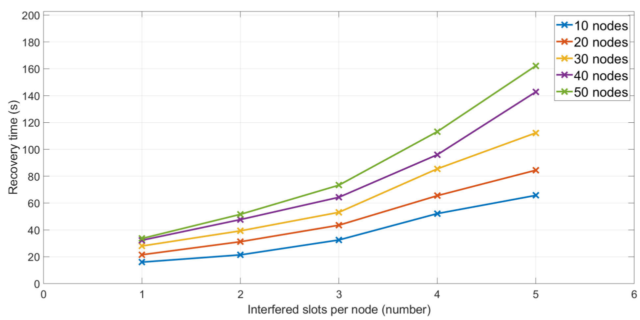

FIGURE 12. Recovery times of the networks with different sizes, operating in a presence of various external interference.

induced by external networks, if their spectrum footprint is not changing rapidly over time.

C. USE CASE 3. MULTI-HOP NETWORKS SPATIAL REUSE AND SCALING EXAMINATION

In an ad hoc network with multi-hop topology, the same time-frequency slots can be reused, due to the spatial separation of nodes in non-overlapping collision domains, where communication links do not interfere with each other. To optimize spatial reuse of spectrum, multi-hop networks must avoid the hidden and exposed node problems that may occur.

To demonstrate the reliable and scalable performance of DDMC-TDMA and its ability to achieve optimal spatial spectrum reuse in the case of multi-hop ad hoc networks, a set of simulations is executed. The number of nodes in the network is varied from 100 to 500 with a step of 100. For every size of the ad hoc network, network density is increased by varying the number of neighbor nodes from 10 to 30 with a step of 5 and random topology graph models of ad hoc networks are generated. Every node in the network communicates with two neighbor nodes: transmitting to one neighbor node and receiving from another neighbor node. The superframe has the same configuration as in the single-hop use case: the number of available channels in the radio spectrum is fixed to 16 and the maximum number of available data slots is 256. However, please note that in a multi-hop network the same slot can be allocated multiple times in different collision domains. A low data rate is adopted and in order to satisfy its application throughput demands, every node is required to allocate 10 slots. This means, that, for instance, in a multi-hop ad hoc network consisting of 100 nodes, 1000 Tx slots must be allocated to accommodate all traffic.

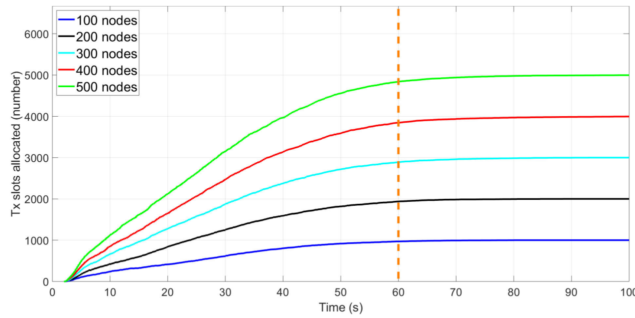

FIGURE 13. Tx slot allocation graph for the multi-hop network topology with the number of neighbor nodes fixed to 20.

Fig. 13 represents an example of allocated Tx slots of the whole multi-hop network over time. The number of neighbor nodes is fixed to 20 for this graph. We observe consistent convergence to steady network state and scalable DDMC-TDMA operation without any degradation of network performance when network size increases, as shown by the linear increase of allocated Tx slots with network size. Fig. 13 further shows that regardless of the network size, 95% of slot allocations is achieved within 60 seconds, after which it converges towards the optimal resource utilization. The same behavior is also observed for simulations with a different number of neighbor nodes (see Fig. 14). The fact that many more than 256 Tx slots can be allocated and steady convergence is achieved, proves the scalable and reliable operation of DDMC-TDMA and the capability to reuse slot spatially. At the same time, hidden and exposed node problems are also avoided, as this is an inherent feature of the protocol design. Occasionally, the hidden node problem might still occur due to parallel execution of the protocol or missed slot usage reports, but this will be resolved by the initiation of a slot removal procedure. Protection against the exposed node problem is described in more detail in the following use case.

Fig. 14 shows the scalability of the protocol in terms of convergence time with increasing network density. Similarly to the single-hop use case, the convergence time increases linearly with network density, indicating that the scalability does not depend on the number of hops in a network. The limiting factor for scalability is the higher control slot utilization and

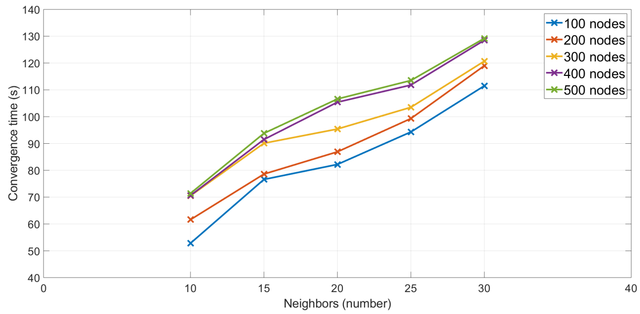

FIGURE 14. Convergence times versus the number of neighbor nodes for various sizes of multi-hop networks.

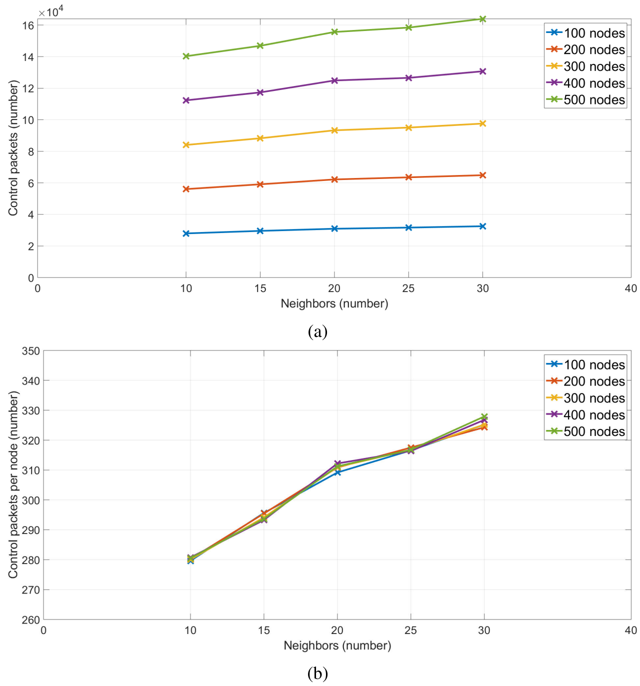

FIGURE 15. Control overhead of multi-hop networks. (a) Overall. (b) Per node.

a higher probability of overlapping slot allocations when the network density increases.

In Fig. 15 control overhead of multi-hop networks with different sizes and densities is presented; overall control overhead of networks in Fig. 15a and per node control overhead in Fig. 15b. With more nodes in the multi-hop networks, more slots need to be allocated, therefore the control overhead is linearly increasing with a linear increase in network size, as shown in Fig. 15a. In addition, control overhead slightly increases with denser wireless deployments, which is more obvious from Fig. 15b. Per node control overhead increases from 280 packets up to approximately 315 packets for network densities of 10 and 30, respectively. Fig. 15b further proves the scalability of the algorithm, as per node control overhead only increases with an increase in network density and is not affected by an increase in network size. It is worth noting that control overhead depends on application demands. If there is no traffic for one or more nodes, there is no exchange of control messages, leading to zero control overhead for nodes in consideration. In executed simulations, traffic demands are constantly present for every node, so this feature of the algorithm is not shown.

D. USE CASE 4. PREVENTION OF EXPOSED NODE PROBLEM IN MULTI-HOP NETWORKS

In the following subsection, the performance and behavior of DDMC-TDMA are analyzed in the case of network topologies and communication patterns that are vulnerable to the exposed node problem. The main focus of the conducted simulations is to validate how DDMC-TDMA is capable to avoid the exposed node problem in order to maximize spatial reuse of the spectrum. The network topology adopted for this use case is presented in Fig. 16.

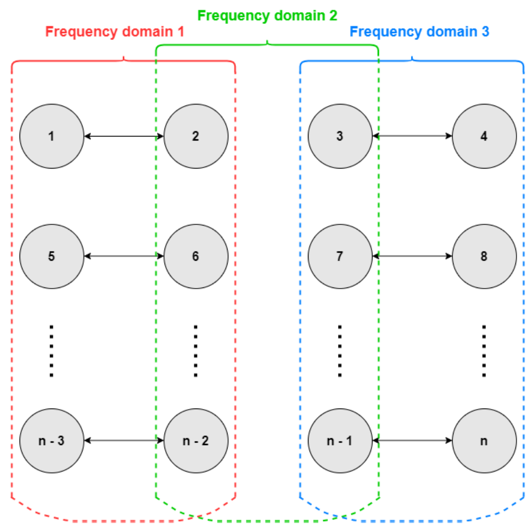

FIGURE 16. Network topology to trigger the exposed node problem.

This network topology is easy to scale up the network in size and allows to compare the performance for different network sizes. The network nodes are divided into 4 groups:

- group 1: nodes 1, 5, …, n-3
- group 2: nodes 2, 6, …, n-2
- group 3: nodes 3, 7, …, n-1
- group 4: nodes 4, 8, …, n,

where n (divisible by 4) is the number of the nodes in an ad hoc network. Nodes from groups 1 and 2 are in the same collision domain and unidirectional data traffic is assumed from one node of group 1 to one node of group 2 or the other way around. The same applies to nodes from groups 3 and 4. Nodes from groups 2 and 3 do not communicate but are in the same collision domain. Such a network topology and point-to-point communication links lead to the occurrence of the exposed node problem.

The exposed node avoidance capability was analyzed for different numbers of frequency channels and different network sizes. The optimal number of channels ($n\_channels$) per simulation is calculated as follows: for a network size of

4 nodes (see Fig. 16), the number of established communication links is 2. For a high data rate, in the optimal case, the two Tx nodes should allocate 16 data slots each. If the proposed protocol offers protection against the exposed node problem, one channel should suffice. For *n* nodes, this generalizes as follows:

$$n\_channels = \frac{n}{4} \tag{3}$$

Optimal spectrum utilization is achieved if every frequency-time slot in the superframe is allocated for any two non-interfering communication links, i.e. one link established between groups 1 and 2 and the other between groups 3 and 4. In this case, the number of allocated Tx slots is:

$$n\_allocated\_tx = n\_channels \cdot 16 \cdot 2 \tag{4}$$

where 16 represents the number of available data slots per channel. If we substitute (3) in (4), then the number of expected allocated Tx slots is:

$$n\_allocated\_tx = n \cdot 8 \tag{5}$$

Unlike most existing distributed scheduling protocols that consider Tx or Rx slots allocated in two-hop neighborhood unavailable for further utilization, the DDMC-TDMA approach detects the presence of exposed nodes and hence avoids spectrum underutilization. As is explained in section III-B1, DDMC-TDMA avoids exposed node problem by keeping track of three different types of slot information, i.e. USED, USED Tx, and USED Rx. If no protection is offered against exposed nodes, the expected number of allocated Tx slots is half of the number calculated in (5).

Variable $n\_allocated\_tx$ represents the optimal number of allocated slots in case that exposed node problem is avoided. To prove the capability of DDMC-TDMA to reach this optimal number of allocated slots, simulations are performed with network size increasing from 4 to 116 nodes in steps of 8. For every simulation, a high application data rate is used. The achieved results meet the expectations: the number of allocated Tx slots is always equal to $n\_allocated\_tx$ for DDMC-TDMA networks (4 nodes - 32 slots, 12 nodes - 96 slots, ..., 116 nodes - 928 slots). As the results are uniform for all simulated network sizes, it is shown that the exposed node avoidance feature of DDMC-TDMA is also scalable with increasing network size.

As previously mentioned in section III-B1, the proposed protocol targets to reuse slots that are already used by the network nodes. In this way, more spectrum is left for external networks operating in the collision domain of the wireless network in consideration. To analyze the practical feasibility in terms of spectrum reuse, simulations are performed using the network topology from Fig. 16 with a varying number of network nodes. In addition to spectrum reuse, these simulations further demonstrate the capability of DDMC-TDMA to avoid exposed node problem. The application data rate adopted for these simulations is a low data rate. For example, in the case of 1 frequency channel available and a network consisting of 4 nodes, it is expected that 10 data slots are allocated and reused by 2 established and non-interfering communication links, while the remaining 6 data slots stay free. Generalizing this example, the optimal number of reused Tx slots can be calculated as:

$$n\_reused\_tx = n\_channels \cdot 10 \tag{6}$$

If we substitute (3) in (6), the optimal number of reused Tx slots is:

$$n\_reused\_tx = \frac{n \cdot 5}{2} \tag{7}$$

Optimal numbers of reused slots and achieved numbers of reused slots in conducted simulations are presented in Table 5. As they represent an average value within a set of 20 runs, presented numbers are not decimal numbers. For every network size, the optimal number of reused slots calculated based on (5) is presented under column 'Reused slots (optimal)'. However, due to asynchronous execution of the protocol and the limited number of slots that can be embedded in the proposed slots message, we can see from column 'Reused slots (achieved)' that achieved reusage of the slots is not reaching the optimal number. Instead of preferred reuse of already allocated slots, it can be seen from the 'Non-reused slots' column that there are slots with only one communication link established. With an increase in network size, there is also an increase in non-reused slots. However, with the increased number of network nodes, there are more available channels in the radio spectrum and more slots to be allocated, thus an increased probability of occurrence of non-reused slots is expected. The best indication of spectrum reusage efficiency can be obtained if we calculate what percentage of optimal reusages is achieved. This percentage is presented in the last column of Table 5. It shows that the percentage-wise difference between the optimal and achieved number of reused slots is kept around the same value, thus no degradation of performance is induced with an increase in network size and an increase in the number of frequency-time slots to be allocated.

TABLE 5. Comparison of the optimal and achieved number of reused slots.

Nodes	Non-reused slots	Reused slots (achieved)	Reused slots (optimal)	Reused slots achieved percentage (achieved·100/optimal) (%)
4	0.2	9.9	10	99.00
12	1.4	29.3	30	97.67
20	1.8	49.1	50	98.20
28	2.8	68.6	70	98.00
36	3.2	88.4	90	98.22
44	2.7	108.7	110	98.82
52	3.4	128.3	130	98.69
60	5.8	147.2	150	98.13
68	5.1	167.6	170	98.59
76	6.9	186.9	190	98.37
84	9.2	205.7	210	97.95
92	8.9	225.7	230	98.13
100	11.0	244.8	250	97.92
108	12.7	264.4	270	97.93
116	14.5	283.5	290	97.76

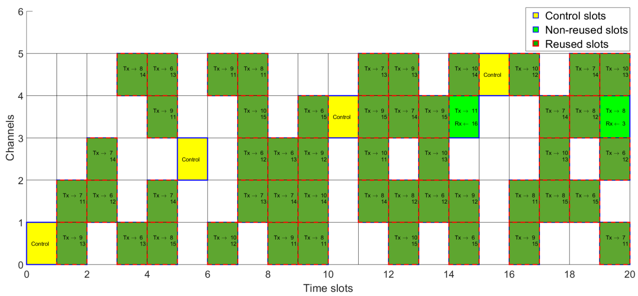

FIGURE 17. Superframe state of 20 nodes network running DDMC-TDMA with achieved near-optimal slot reusage.

An example of spectrum utilization for 20 nodes network is presented in Fig. 17. Only in two time-frequency slots, single communication links are established. In an ideal case, these two links would occupy the same slot, leaving one more slot for external networks. All other slots are successfully reused by two communication links.

V. CONCLUSION

In this paper, we have proposed a novel medium access protocol, called dynamic distributed multi-channel TDMA (DDMC-TDMA), that is capable to address the main shortcomings in wireless ad hoc networks, such as scalability, spectral efficiency, QoS, and coexistence of multiple co-located ad hoc networks. DDMC-TDMA was initially used and experimentally validated in the DARPA SC2 competition, where its flexibility and reliability in various challenging wireless scenarios for dynamic spectrum sharing with other unknown wireless technologies has been successfully proven.

In this paper, the detailed architectural design of the DDMC-TDMA protocol and its modules has been presented. The central module is the scheduling table with different types of internally and externally used slots that are maintained per node. The architecture further includes procedures and associated control messages for slot allocation and removal operations. The exchange of control messages happens via dedicated control slots in a medium access scheme. The DDMC-TDMA protocol has been validated through ns-3 simulations for various single-hop and multi-hop wireless scenarios and it is shown that the protocol is scalable in all scenarios, without performance degradation.

For single-hop networks, convergence times are proportional to the number of nodes (varied between 20 and 50) and the number of slots available for resource allocation. The time needed for achieving above 95% spectrum utilization is in the range of 55-100 seconds for basic DDMC-TDMA and in the range of 50-70 seconds for DDMC-TDMA with the optimization of randomized initiation of slot removal operations. After the convergence time, single-hop networks are able to converge to a steady state and fully utilize the available radio spectrum. The slot allocation mechanism has proven to be reliable and scalable for different network densities within the single-hop collision domain. It has also been shown that with small extensions on top of basic DDMC-TDMA, network performance can be further improved.

The performance of DDMC-TDMA has also been evaluated in presence of multiple co-located and unknown external networks. The size of the DDMC-TDMA based network is varied from 10 to 50 nodes, with each node utilizing 16 Tx slots and with external networks causing interference in 1 up to 5 Tx slots per node. Depending on the number of concurrently interfered slots, the recovery time, being the time needed until all affected slots are re-allocated and a steady state is achieved again, is ranging between 18 and 160 seconds. It has been shown that with DDMC-TDMA, networks of different sizes and densities fully recover in the presence of different interference levels. This proves that DDMC-TDMA by design can coexist with unknown external networks operating in the same collision domain.

In the multi-hop network, regardless of the network size, 95% spectrum utilization is achieved within 60 seconds. All links are able to allocate sufficient slots to satisfy application demands by maximally exploiting spatial reuse of time-frequency slots. As a result, the spectrum footprint of ad hoc networks can be minimized, meaning that spectrum efficiency is optimized, hence preserving more spectrum for other networks operating in the same collision domain. Networks running DDMC-TDMA are able to avoid both hidden and exposed node problems leading to near-optimal spectrum utilization and improved QoS guarantees. For network topologies suffering from the exposed node problem, this problem is fully mitigated, regardless of network size. At the same time, spatial reuse is achieved for around 98% of slots.

The DDMC-TDMA protocol can be applied to any wireless network that requires high-density deployment and needs to adapt to a dynamic environment while maintaining high reliability, high spectral efficiency, and seamless coexistence with external networks. In summary, the proposed DDMC-TDMA protocol offers an all-in-one solution tackling by design many well-known problems only partially addressed by other existing distributed TDMA scheduling protocols.

REFERENCES

[1] Y. Song and J. Xie, "BRACER: A distributed broadcast protocol in multi-hop cognitive radio ad hoc networks with collision avoidance," *IEEE Trans. Mobile Comput.*, vol. 14, no. 3, pp. 509–524, Mar. 2015.

[2] M. Hadded, P. Muhlethaler, A. Laouiti, R. Zagrouba, and L. A. Saidane, "TDMA-based MAC protocols for vehicular ad hoc networks: A survey, qualitative analysis, and open research issues," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2461–2492, Oct./Dec. 2015.

[3] W. Zafar and B. M. Khan, "Flying ad-hoc networks: Technological and social implications," *IEEE Technol. Soc. Mag.*, vol. 35, no. 2, pp. 67–74, Jun. 2016.

[4] S. Sharmila and T. Shanthi, "A survey on wireless ad hoc network: Issues and implementation," in *Proc. Int. Conf. Emerg. Trends Eng., Technol. Sci. (ICETETS)*, Feb. 2016, pp. 1–6.

[5] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proc. IEEE 21st Annu. Joint Conf. Comput. Commun. Soc. (INFOCOM)*, vol. 3. Jun. 2002, pp. 1567–1576.

[6] I. Rhee, A. Warrier, M. Aia, J. Min, and M. L. Sichitiu, "Z-MAC: A hybrid MAC for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 511–524, Jun. 2008.

[7] D. J. Vergados, D. D. Vergados, and M. Koutsogiannaki, ''Capacity optimization in TDMA ad-hoc networks,'' *Wireless Pers. Commun.*, vol. 62, no. 3, pp. 687–713, Feb. 2012.

[8] M. Yu, C. Zhou, and M. Zhao, ''De-jitter dynamic slot allocation algorithm based on cluster analysis,'' *Comput. Sci.*, vol. 44, no. 7, pp. 94–97, Jul. 2017.

[9] Y. Li, X. Zhang, J. Zeng, Y. Wan, and F. Ma, ''A distributed TDMA scheduling algorithm based on energy-topology factor in Internet of Things,'' *IEEE Access*, vol. 5, pp. 10757–10768, 2017.

[10] M. A. Sheikh, N. B. Z. Ali, and A. Awang, ''Distributive and self-sustainable scheduling algorithm for wireless sensor networks,'' in *Proc. 22nd Wireless Opt. Commun. Conf.*, May 2013, pp. 415–420.

[11] D. Helen and D. D. Arivazhagan, ''Applications, advantages and challenges of ad hoc networks,'' *J. Acad. Ind. Res.*, vol. 2, no. 8, pp. 453–457, Jan. 2014.

[12] S. Chen and J. Zhao, ''The requirements, challenges, and technologies for 5G of terrestrial mobile telecommunication,'' *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 36–43, May 2014.

[13] K.-H. Phung, B. Lemmens, M. Goossens, A. Nowe, L. Tran, and K. Steenhaut, ''Schedule-based multi-channel communication in wireless sensor networks: A complete design and performance evaluation,'' *Ad Hoc Netw.*, vol. 26, pp. 88–102, Mar. 2015.

[14] A. Jayasuriya, S. Perreau, A. Dadej, and S. Gordon, ''Hidden vs. Exposed terminal problem in ad hoc networks,'' in *Proc. Aust. Telecommun. Netw. Appl. Conf.*, 2004, pp. 52–59.

[15] Defense Advanced Research Project Agency (DARPA). *Spectrum Collaboration Challenge (SC2)*. Accessed: Dec. 8, 2020. [Online]. Available: https://archive.darpa.mil/sc2/

[16] S. Giannoulis, M. Mehari, A. Shahid, D. Stojadinovic, M. Claeys, F. Mahfoudhi, W. Liu, I. Seskar, S. Latre, I. Moerman, C. Donato, R. Mennes, F. A. P. D. Figueiredo, I. Jabandzic, Y. D. Bock, M. Camelo, J. Struye, and P. Maddala, ''Dynamic and collaborative spectrum sharing: The SCATTER approach,'' in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Nov. 2019, pp. 1–6.

[17] NS-3 Consortium. *NS-3 Network Simulator*. Accessed: Dec. 8, 2020. [Online]. Available: https://www.nsnam.org/

[18] C. Zhu and M. S. Corson, ''A five-phase reservation protocol (FPRP) for mobile ad hoc networks,'' in *Proc. Conf. Comput. Communications. 17th Annu. Joint Conf. IEEE Comput. Commun. Societies. Gateway 21st Century*, vol. 1, Mar. 1998, pp. 322–331.

[19] Y. Li, X. Zhang, T. Qiu, J. Zeng, and P. Hu, ''A distributed TDMA scheduling algorithm based on exponential backoff rule and energy-topology factor in Internet of Things,'' *IEEE Access*, vol. 5, pp. 20866–20879, 2017.

[20] I. Rhee, A. Warrier, J. Min, and L. Xu, ''DRAND: Distributed randomized TDMA scheduling for wireless ad hoc networks,'' *IEEE Trans. Mobile Comput.*, vol. 8, no. 10, pp. 1384–1396, Oct. 2009.

[21] S. Ramanathan, ''A unified framework and algorithm for (T/F/C)DMA channel assignment in wireless networks,'' in *Proc. INFOCOM*, vol. 2, Apr. 1997, pp. 900–907.

[22] K. Sato and S. Sakata, ''A distance-measurement-oriented distributed TDMA scheduling algorithm for sensor networks,'' in *Proc. Int. Conf. Distrib. Comput. Sensor Syst. Workshops (DCOSS)*, Jun. 2011, pp. 1–3.

[23] K. L. Bryan, T. Ren, T. Henry, and V. Fay-Wolfe, ''Towards optimal TDMA frame size in wireless sensor networks,'' Dept. Comput. Sci. Statist., Univ. Rhode Island, Kingston, RI, USA, Tech. Rep., 2007.

[24] B. Dezfouli, M. Radi, K. Whitehouse, S. Abd Razak, and T. Hwee-Pink, ''DICSA: Distributed and concurrent link scheduling algorithm for data gathering in wireless sensor networks,'' *Ad Hoc Netw.*, vol. 25, pp. 54–71, Feb. 2015.

[25] D. J. Vergados, N. Amelina, Y. Jiang, K. Kralevska, and O. Granichin, ''Toward optimal distributed node scheduling in a multihop wireless network through local voting,'' *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 400–414, Jan. 2018.

[26] *IEEE Standards 802.15.4: Wireless Medium Access Control (MAC) and Physical layer (PHY) specification for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Standard 802.15 WPAN Task Group, Sep. 2006.

[27] *IEEE Standard for local and Metropolitan Area Networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer*, IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011), Apr. 2012, pp. 1–225.

[28] P. Thubert, *An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4*, document Internet Eng. Task Force, Internet-Draft draft-ietf-6tisch-architecture-30, Nov. 2020, Work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-6tisch-architecture-30

[29] Q. Wang and X. Vilajosana, *6TiSCH Operation Sublayer (6top)*, document Internet Engineering Task Force, Internet-Draft draft-wang-6tisch-6top-sublayer-04, Nov. 2015, Work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-wang-6tisch-6top-sublayer-04

[30] Q. Wang, X. Vilajosana, and T. Watteyne, *6TiSCH Operation Sublayer (6top) Protocol (6P)*, document RFC 8480, Nov. 2018. [Online]. Available: https://rfc-editor.org/rfc/rfc8480.txt

[31] D. Dujovne, L. A. Grieco, M. R. Palattella, and N. Accettura, *6TiSCH 6top Scheduling Function Zero (SF0)*, document Internet Engineering Task Force, Internet-Draft draft-ietf-6tisch-6top-sf0-05, Jul. 2017, Work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-6tisch-6top-sf0-05

[32] D. Dujovne, L. A. Grieco, M. R. Palattella, and N. Accettura, *6TiSCH Experimental Scheduling Function (SFX)*, document Internet Engineering Task Force, Internet-Draft draft-ietf-6tisch-6top-sfx-01, Mar. 2018, Work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-6tisch-6top-sfx-01

[33] T. Chang, M. Vucinic, X. Vilajosana, S. Duquennoy, and D. Dujovne, *6TiSCH Minimal Scheduling Function (MSF)*, document Internet Engineering Task Force, Internet-Draft draft-ietf-6tisch-msf-18, Sep. 2020, Work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-6tisch-msf-18

[34] S. Kharb and A. Singhrova, ''A survey on network formation and scheduling algorithms for time slotted channel hopping in industrial networks,'' *J. Netw. Comput. Appl.*, vol. 126, pp. 59–87, Jan. 2019.

[35] A. Aijaz, A. Stanoev, and U. Raza, ''GALLOP: Toward high-performance connectivity for closing control loops over multi-hop wireless networks,'' in *Proc. 27th Int. Conf. Real-Time Netw. Syst. (RTNS)*, 2019, pp. 176–186.

[36] S. M. Kamruzzaman and M. S. Alam, ''Dynamic TDMA slot reservation protocol for cognitive radio ad hoc networks,'' in *Proc. 13th Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2010, pp. 142–147.

[37] S. Agarwal, R. K. Shakya, Y. N. Singh, and A. Roy, ''DSAT-MAC: Dynamic slot allocation based TDMA MAC protocol for cognitive radio networks,'' in *Proc. 9th Int. Conf. Wireless Opt. Commun. Netw. (WOCN)*, Sep. 2012, pp. 1–6.

[38] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie, ''Protocols for self-organization of a wireless sensor network,'' *IEEE Pers. Commun.*, vol. 7, no. 5, pp. 16–27, Oct. 2000.

[39] R. Nelson and L. Kleinrock, ''Spatial TDMA: A collision-free multihop channel access protocol,'' *IEEE Trans. Commun.*, vol. 33, no. 9, pp. 934–944, Sep. 1985.

[40] D. Rodenas-Herraiz, F. Garcia-Sanchez, A.-J. Garcia-Sanchez, and J. Garcia-Haro, ''On the influence of the hidden and exposed terminal problems on asynchronous IEEE 802.15.5 networks,'' *Comput. Standards Interfaces*, vol. 42, pp. 53–70, Nov. 2015.

[41] R. T. B. Ma, V. Misra, and D. Rubenstein, ''An analysis of generalized slotted-aloha protocols,'' *IEEE/ACM Trans. Netw.*, vol. 17, no. 3, pp. 936–949, Jun. 2009.

[42] I. Jabandžić. (Jan. 2020). *A Dynamic Distributed Multi-Channel TDMA (DMC-TDMA) Scheduling Algorithm Implementation for NS-3*. [Online]. Available: https://github.com/IrfanJabandzic/MfTDMA-ns-3

[43] F. A. P. D. Figueiredo, D. Stojadinovic, P. Maddala, R. Mennes, I. Jabandzic, X. Jiao, and I. Moerman, ''Scatter phy: A physical layer for the DARPA spectrum collaboration challenge,'' in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Nov. 2019, pp. 1–6.

IRFAN JABANDŽIĆ received the B.Sc. and M.Sc. degrees in automatic control and electronics from the Faculty of Electrical Engineering Sarajevo, Bosnia and Herzegovina, in 2012 and 2014, respectively. He is currently pursuing the Ph.D. degree with Ghent University. After working in the area of embedded engineering for two years, in October 2016, he joined the Internet Technology and Data Science Laboratory, Ghent University. He was involved in the DARPA Spectrum Collaboration Challenge (SC2) as a MAC specialist for the SCATTER team and participated in the European Space Agency CODYSUN project, working on MAC layer and dynamic spectrum sharing techniques. He contributed towards ten publications in his field of research. His main research interests include wireless networks and wireless sensor networks, focusing on flexible and adaptive MAC architectures, heterogeneous wireless technologies coexistence, and dynamic spectrum access.

SPILIOS GIANNOULIS (Member, IEEE) received the Diploma degree in electrical and computer engineering, in 2001, and the Ph.D. degree in the area of routing for wireless mobile sensor networks, in 2010. From 2001 to 2015, he was involved in several E.U. and national research and development projects at Industrial systems Institute and the Applied Electronics Laboratory (APEL), University of Patras, Greece. From 2005 to 2015, he was a Lecturer with the Technological Educational Institute of Patras, Greece. He is currently a Senior Researcher with the IMEC, Belgium, involved in several EU projects in the area of reconfigurable and adaptive wireless networks. He has served as a Technical Leader of the H2020 project WiSHFUL and of the Scatter team that took part in the DARPA Second Spectrum Collaboration Challenge, resulting in winning 1.5 million U.S. dollars prizes in total from DARPA. He is also involved in the European Space Agency CODYSUN project. His main research interests include wireless networks, mobile ad-hoc networks, wireless sensor networks, focusing on flexible and adaptive MAC and routing protocols, QoS provisioning, cross-layer and power aware system architecture design, spectrum resource management for public and licensed bands, dynamic spectrum access, and 4G-5G networks with more than 40 relevant publications in international conferences and journals.

RUBEN MENNES (Associate Member, IEEE) received the master's degree in computer science from the University of Antwerp, Belgium, in June 2016. He is currently pursuing the Ph.D. degree with the University of Antwerp—IMEC. He joined the Department of Mathematics and Computer Science, University of Antwerp and IMEC, in August 2016. The focus of his research is mainly on collaborative wireless networks and smart wireless networks, the cross-domain between wireless network technologies and machine learning. He was part of the SCATTER team of the DARPA SC2 competition, where he focused on the structure, design, implementation, and research of the decision making and optimization engine.

FELIPE A. P. DE FIGUEIREDO received the B.S. and M.S. degrees in telecommunications from the Instituto Nacional de Telecomunicações (INATEL), Minas Gerais, Brazil, in 2004 and 2011, respectively. He is currently pursuing the Ph.D. degree with the Internet Technology and Data Science Laboratory, Ghent University, Gent, Belgium. He has been working with the research and development of telecommunications systems for more than ten years. His research interests include digital signal processing, digital communications, mobile communications, MIMO, multicarrier modulations, and FPGA development.

MAXIM CLAEYS received the master's degree in computer science and the Ph.D. degree in Quality of Experience optimization for multimedia delivery from Ghent University, Belgium, in June 2012. He joined the Department of Information Technology, Ghent University, in August 2012. After obtaining his Ph.D. degree with his dissertation titled Quality-Driven Management of Video Streaming Services in Segment-Based Cache Networks, in March 2017, he joined the IDLab Research Group, University of Antwerp, where he was actively holding a postdoctoral position in the area of artificial intelligence. He joined Robovision, as a Project Engineer, in September 2018.

INGRID MOERMAN (Member, IEEE) received the degree in electrical engineering and the Ph.D. degree from Ghent University, in 1987 and 1992, respectively. She became a part-time Professor with Ghent University, in 2000. She is currently a Staff Member with IDLab, a Core Research Group of IMEC with research activities embedded with Ghent University and University of Antwerp. She is coordinating the research activities on mobile and wireless networking at Ghent University, where she is leading a research team of more than 30 members. She is also a Program Manager of the Deterministic Networking track at IMEC and in this role she coordinates research activities on end-to-end wired/wireless networking solutions driven by time-critical applications that have to meet strict QoS requirements in terms of throughput, latency bounds and dependability in smart application areas like Industry 4.0, vehicular networks, and professional entertainment. Her main research interests include cooperative and intelligent radio networks, real-time software defined radio, time-sensitive networks, dynamic spectrum sharing, coexistence across heterogeneous wireless networks, vehicular networks, open-source prototyping platforms, software tools for programmable networks, next generation wireless networks (5G/6G/...), and experimentally supported research. She has a longstanding experience in running and coordinating national and EU research funded projects. At the international level, she was leading team SCATTER, consisting of researchers from IMEC-IDLab and Rutgers University (USA), in the DARPA Spectrum Collaboration Challenge (SC2). The SCATTER team was one of the ten finalists at the DARPA SC2 championship event organized at Mobile World Congress, LA, in October 2019.