A Reservation-based Adaptive MAC Protocol for OFDM Physical Layers in Underwater Networks

Sara Falleni, Tommaso Melodia, Stefano Basagni

The Institute for the Wireless Internet of Things, Northeastern University, Boston, USA

Abstract—The more we understand the importance of the oceans for human well-being and survival, the more research on the *Internet of Underwater Things* becomes imperative. This prompts us to investigate technologies for underwater communication and networking, namely, technologies that enable the collection of data that are vital for "Blue Economy" applications. Among these technologies, Orthogonal Frequency-Division Multiplexing (OFDM)—extensively used in terrestrial networks— is being considered because of its high spectral efficiency, low inter-symbol interference and fading, and low sensitivity to time synchronization errors. In this paper, we investigate how OFDM physical layers affect protocol design and performance at higher layers of the protocol stack. Particularly, we present a Reservation-based Adaptive MAC (RAMAC) protocol that leverages the capabilities of OFDM physical layers. Using information about channel conditions, RAMAC selects the OFDM frequencies to use on a per-packet basis. We evaluate the performance of RAMAC via DESERT-based simulations on a variety of underwater scenarios with models of real OFDM-enabled underwater acoustic modems. Results show that, especially when outside interference is present (e.g., sonars), RAMAC provides robust data delivery while keeping latency at bay.

I. INTRODUCTION

As we become increasingly aware that the well-being of our oceans and waterways is paramount for life on Earth, our interest in studying and understanding them grows. Our knowledge about the oceans relies mainly on the amount and quality of the information that we are able to retrieve from them. Such information can be harvested in different ways: through sensors placed on the bottom of the oceans, floating buoys, underwater vehicles, etc. For continued information harvesting and for relaying it to the shore promptly, marine devices must be able to communicate with each other and form a network, which most of the time needs to be wireless so that mobility and greater extent can be accommodated. As a consequence, research concerning the so-called *Internet of Underwater Things* is intensifying [1]. Its main enablers are *underwater modems* that allow the exchange of data wirelessly. Particularly, modems using acoustic technology are to be used as only sound propagates in the water for hundreds of meters, while electromagnetic waves do not. To constantly exchange information, underwater acoustic modems must be able to adapt and work in the sea, which is challenging, especially for modems constrained to use single frequencies. This is why recent underwater acoustic communication devices are designed to use the Orthogonal Frequency-Division Multiplexing (OFDM) technology at the physical layer. OFDM offers high spectral efficiency, low inter-symbol interference and

fading, and low sensitivity to time synchronization errors [2]. Moreover, OFDM partitions the bandwidth into subcarriers that can be used independently, allowing greater adaptability and resistance to interference external to the network (e.g., a sonar or other noise from a passing vessel).

The use of OFDM in underwater networks is an active research area. Although initially focused mostly on communications [3], using OFDM for protocols at higher layers of the protocol stack is gaining attention. Zhang et al. propose an OFDM-based frequency division multichannel media access (OFDM-FDMMA) for data collection [4]. The protocol uses channel state information statistics to explore multi-user frequency diversity. Zhou et al. propose a MAC protocol that uses Time Division Multiplex (TDM) with Frequency Division Multiplex (FDM) on top of OFDM (TFO-MAC) [5]. TDM with FDM/OFDM are jointly used for uplink traffic in a cellular-like underwater acoustic network. As such, TFO-MAC is tested in one-hop scenarios where nodes communicate with the sink only. Yin et al. provide another example of OFDM-based underwater MAC that is amenable to single-hop networks [6]. Exploiting information from the physical layer, their protocol chooses modulation, coding, and frequency diversity to achieve variable rate transmission. Carrier Sensing Multiple Access with Collision Avoidance (CSMA/CA) mechanism is used to decrease interference. Finally, Su et al. propose an OFDMA-based Subcarriers Pregrouping MAC protocol called OSPG-MAC for underwater acoustic wireless sensor networks [7]. OSPG-MAC assigns different subcarriers to neighboring nodes allowing them to perform simultaneous transmission while avoiding collisions. An example of an underwater modem featuring an OFDM physical layer has been built by the *SEANet Project* [8], [9]. Through the use of OFDM, the SEANet modem is capable of supporting data rates about one order of magnitude higher than those of existing commercial platforms over short to moderate-range links.

Contribution. While OFDM-based devices are being developed, it's important to investigate how OFDM-based technologies affect communication and networking in different underwater scenarios. Specifically, in this paper we present a *Reservation-based Adaptive MAC* (RAMAC) protocol that, using OFDM at the physical layer, uses knowledge of the channel conditions to determine the subcarriers to be used to avoid interference. RAMAC operations are based on a simple channel reservation mechanism that is also used to communicate which subcarriers should be used for data trans-

mission on a per-packet basis. We implement the protocol in the open-source underwater simulator *DESERT* [10], [11], which we chose for its widespread use, and for which we developed a SEANet-based OFDM physical layer module [12]. We evaluate the performance of RAMAC against a baseline single carrier protocol, challenging them in various scenarios. Our results show that RAMAC offers superior Packet Delivery Ratio (PDR) while keeping the latency low. Particularly, when the channel is affected by strong interference, RAMAC outperforms the baseline solution by 20% for the PDR while achieving end-to-end latency up to three orders of magnitude smaller. This suggests that, as for terrestrial radio networks, leveraging OFDM capabilities for channel access leads to more robust data delivery and overall better performance.

The rest of the paper is organized as follows. In Section II we provide a description of the network scenario where our protocol operates. The operations of RAMAC are detailed in Section III. Section IV illustrates results from the performance evaluation of RAMAC. Finally, Section V concludes the paper.

II. NETWORK SCENARIO

We consider static underwater wireless sensor networks (UWSNs) with N nodes. Each node is equipped with one or more sensors. Sensors produce data to be delivered to the network data collection point (the *sink*), for processing and further forwarding. The sink is located a few meters under the sea surface, in the center of the deployment area, which is enclosed in a rectangular cuboid $L \times W \times D$, with the upper face coinciding with the sea surface. The N nodes are randomly deployed within this area. Because of the extent of node deployment and underwater channel dynamics, not all nodes can directly communicate with the network sink, i.e., the network topology is multi-hop. We stipulate that data packets follow pre-defined routes (e.g., shortest paths) from source to sink. Fig. 1 depicts a sample UWSN multi-hop scenario.
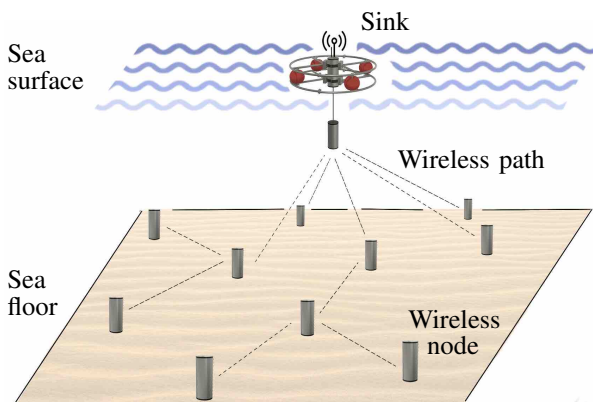

Fig. 1: An underwater wireless sensor network scenario.

For data transmission, each node is equipped with a half-duplex omni-directional acoustic modem that uses an OFDM physical layer. In particular, we focus on scenarios comprising last-generation software-defined acoustic modems like the SeaNET modem, which achieve higher data rates within short/medium distances [8], [13] and that, given the small form factor and light weight, offers convenient deployment [9]. A desirable characteristic of these modems (when using OFDM) is being able to receive multiple packets transmitted on different subcarriers at the same time. The experimental evaluation offered in this paper leverages DESERT modules developed after the SeaNET modem design [12].

III. THE RAMAC PROTOCOL

At its core, RAMAC is a channel access mechanism for OFDM acoustic physical layers. This means that accessing the channel starts with selecting suitable OFDM subcarriers to be used for transmission. In RAMAC, this is done through a *subcarrier reservation mechanism* based on Ready-To-Send (RTS)/Clear-To-Send (CTS) messages as generally defined by prevailing wireless technologies (e.g., Wi-Fi). Packets awaiting transmission are queued. RTS and CTS control messages are used by nodes with enqueued packets to determine which subcarriers should be used for transmitting those packets. Particularly, with an RTS the sender requests the receiver to indicate which subcarriers it should use to transmit the packets. In the request packet, the sender includes its available subcarriers and the number of bytes that it would like to send. The receiver replies with the list of those subcarriers to be used, namely, those that the receiver senses idle at this time, and the amount of time allocated to the sender. The reservation mechanism is *out of band*, in that RTS and CTS are exchanged in reserved subcarriers, never used for data. The subcarrier reservation mechanism is key for a dynamic multi-carrier system to succeed, as it provides the receiver with real-time information on which subcarriers should be processed when a signal is received, namely, what part of the bandwidth should be considered.

To implement subcarrier reservation, each node keeps a local *Occupancy Table* to track subcarriers usage. The circular bi-dimensional table records which subcarriers are busy and for how long (the duration of transmission, is contained in the RTS and CTS messages). The Occupancy Table is updated (i) when a CTS is sent for upcoming data exchange, (ii) when a CTS concerning another data exchange is overheard, and, in general, (iii) when part of the bandwidth is considered busy (like when interfering signals external to the networks are heard). A list of subcarriers that should not be used, called *No Use Carriers*, is also kept by each node. These subcarriers are marked as *busy* in the Occupancy Table. Each node also keeps track of signals that are received from external nodes, i.e., devices that are not part of its network. Examples include sonars of nearby ships or packets from nodes in other networks. Every time a new unrecognized packet is received, RAMAC, interacting with the physical layer, learns which subcarriers are used to transmit the packet. It then checks whether the same subcarriers were already used by other unrecognized packets in the last n seconds. In the positive case, there is likely a prolonged interference in that part of the channel, and therefore these subcarriers are added to the *No Use Carriers* list and marked as *busy* in the Occupancy Table.

Before sending an RTS or a CTS, a node checks its Occupancy Table: If all data subcarriers are busy, the node awaits a time `free_t`, after which it checks again if at least one of them is free. When a node is not transmitting or receiving, it checks if any RTS was recently received but not processed. If that is the case, the node processes it.

A. RAMAC operations

The RAMAC workflow is exemplified in Fig. 2. In the figure, a white circle indicates that RAMAC will continue the operation that it was performing before the triggering event.
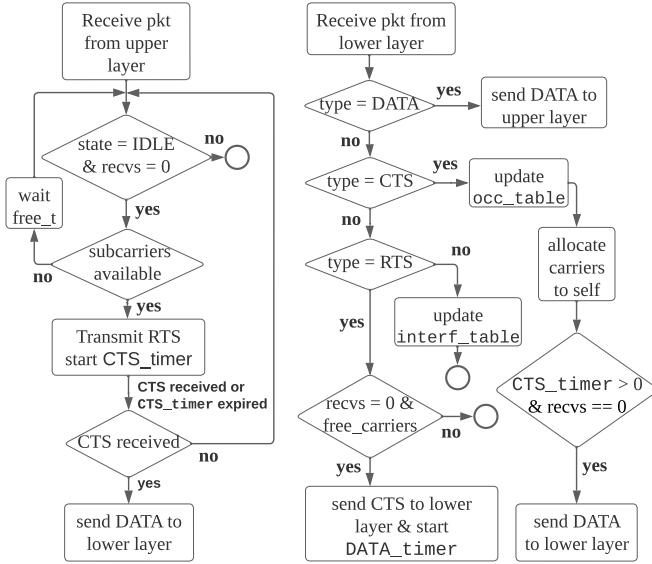

Fig. 2: RAMAC transmission (left) and reception (right).

Nodes running RAMAC begin their life cycle in an IDLE state and are triggered by two major events: the reception of a packet from the upper layer or the reception of a packet from the lower layer. Other relevant events are the end of a transmission and the end of a scheduled timer.

Received packet from upper layer. When data is received from the upper layer, if the node is busy or all subcarriers in the Occupancy Table are busy, it stores the packet in a queue. Otherwise, it sends an RTS and waits to receive a CTS.

Received packet from lower layer. After the reception of a data packet RAMAC sends it to the upper layer and becomes IDLE. After the reception of an RTS, a node awaits for all ongoing receptions to end, if any; then, if there are available subcarriers in the Occupancy Table, the node sends a CTS. After the reception of a CTS, a node selects the subcarriers specified in the CTS, awaits for all receptions to finish, if any, and then sends one or more queued data packets.

End of transmission. When a node ends a transmission, if the transmitted packet was a data packet, it goes back to being IDLE. If instead, the packet was an RTS the node waits for a CTS for a `CTS_timer` time. If a CTS is not received within `CTS_timer` time and if no receptions are ongoing, the node sends another RTS; otherwise goes back to the IDLE state.

If the packet was a CTS, the node awaits a data packet for a `DATA_timer` time. If a data packet is not received within that time, the node goes back to being IDLE.

Timers. Timers are used to perform operations such as waiting for a CTS or a data packet. When an RTS is sent, the `CTS_timer` is started. If the timer expires, the CTS is considered lost and the node tries to send a new RTS. When a CTS is sent a data packet is awaited; if the timer goes off, the packet is considered lost, and the node is free to perform other operations. When a node needs to send an RTS but all the subcarriers are busy the node waits for a time `free_t` before checking the subcarriers again. To make up for the time lost in the RTS/CTS handshake, RAMAC allows nodes to request the channel for a time long enough to transmit up to 5 packets. This number appears to be a good tradeoff between time-saving and avoiding to deny the channel to other nodes.

IV. PERFORMANCE EVALUATION

We demonstrate the effectiveness of RAMAC in providing swift and robust channel access via simulations in different scenarios. We implement RAMAC and other single-carrier MAC protocols in the simulator *DESERT Underwater*, or simply DESERT, a complete set of public C++ libraries to test the design and implementation of underwater network protocols [10]. The performance of RAMAC is compared to that of ALOHA, CSMA, CSMA/CA, and T-Lohi. In this paper, we only show the comparison with CSMA, as in the investigated scenarios this MAC protocol outperforms all other single-carrier protocols. The DESERT implementation of RAMAC relies on OFDM Physical Layer and OFDM-based Interference modules of our design [12].

A. Underwater Scenarios

We consider a multi-hop network where $N = 20$ nodes generate and transmit packets to the sink. The nodes are statically positioned in a cubic area of $140 \times 140 \times 70$ cubic meters according to a uniform random variable. The sink node sits at the center of the deployment area's top face. The actual modem is 10 meters below the surface tethered to a floating buoy that allows it to communicate with land (see Fig. 1 and [9]). Paths from each node to the sink are determined using the Dijkstra algorithm. The cost of each link is the distance between the two nodes.

The nodes and the sink are equipped with an acoustic device modeled after the SEANet modem [8], which allows them to exchange packets. Consistent with the SEANet modem design, each node uses a $125\,\text{kHz}$ center frequency and a $125\,\text{kHz}$ bandwidth. As the nominal SEANet modem transmission range is below $100\,\text{m}$, in our experiments two nodes can communicate directly with each other if their distance is ≤ 70 meters. In our experiments, the underwater acoustic waves propagate at $1500\,\text{m/s}$. Data packets are generated in time according to Poisson distributions with several values of the mean (represented by λ). Particularly, we ran scenarios with $\lambda \in \{0.1, 0.62, 1.25, 1.87, 2.50\}$. The theoretical spectral efficiency of devices using OFDM (S_{OFDM}) is expected to

be twice the spectral efficiency of devices using single carrier modulations (S_{SC}) [14]. More often, in practice, is $S_{OFDM} \sim 1.75 \times S_{SC}$. As such, when communicating over the full bandwidth, nodes using RAMAC over OFDM communicate at 64000 bps (bits per second) while nodes using CSMA over a single carrier communicate at 36000 bps. Each subcarrier uses Binary Phase-Shift Keying (BPSK) modulation.

In our scenarios, we consider an interfering device external to the network (e.g., a sonar or another acoustic transmitter). We show results for interfering pulses generated every 1, 0.5, and 0.25 seconds. Each pulse lasts 0.125 seconds. Our simulations concern data packets of $384\,\mathrm{B}$ or $1536\,\mathrm{B}$.

The performance of the considered protocols is evaluated by investigating the following two metrics.
Packet Delivery Ratio (PDR), defined as the percentage of packets that are successfully received by the sink.
End-to-end latency, defined as the average time to deliver a packet to the sink successfully.

Each simulation run lasts $5000\,\mathrm{s}$. Results are obtained by averaging the outcome of a number of simulations that is enough to achieve 95% confidence and 5% precision.

B. RAMAC Implementation Details

RAMAC uses $14\,\mathrm{B}$ long RTS and $20\,\mathrm{B}$ long CTS control packets. The CTS packet is longer than the RTS packet as it carries the selected subcarriers along with the time allocated to them. Bursts of up to 5 packets can be requested by the sending node and cleared by the receiving node. As such, a single RTS/CTS exchange can be followed by up to 5 data packets. For each burst of packets, the maximum number of RTS that are sent is 4; after the fourth attempt, the packets are dropped. In the OFDM Physical Layer, the bandwidth is divided into 10 subcarriers that can be used independently, numbered 0 through 9. Subcarriers 0 and 1 are used for control packets; the others are used for data packets. Nodes choose data subcarriers on a per-transmission basis. We stipulate that the external interfering device communicates over the frequency range corresponding to subcarrier 5.

C. DESERT's CSMA

We use DESERT's implementation of CSMA to benchmark the performances of RAMAC. DESERT's CSMA operates as follows: When a packet is received from the upper layer, if the MAC layer is idle it starts listening to the channel. The node listens to the channel for a random time $t \in [0.1, r + 0.1]$ where r is a randomly generated number such that $r \in [0, 0.5]$; if the listen timer expires without packets being detected, then the data packet is transmitted, after which the node awaits for the corresponding acknowledgment packet (ACK). If the ACK is not received, the data packet is re-transmitted a maximum of 4 times. The backoff time bt between subsequent trials is $bt \in [0, r \times 2 \times 5 \times 2^c]$, where c is the number of re-transmissions and $r \in [0, 1]$ is a randomly generated number. Data and ACK frames are transmitted on the channel using the whole bandwidth. ACK packets are 10 Bytes long.

D. Performance Results for 384 Byte Packets

Results of simulations in scenarios with $384\,\mathrm{B}$ long packets are shown in Fig. 3 and Fig. 4.

1) Packet delivery ratio: Results concerning the Packet Delivery Ratio can be seen in Fig. 3. From left to right, the interfering node is initially absent, and then it is introduced with an increasingly shorter pulse period. In Fig. 3a (no interference) CSMA has roughly 100% PDR for all traffic classes, with a less than 1% drop for $\lambda = 2.50$. RAMAC has a comparable performance with the lowest traffic but its PDR%, while remaining over 92% even for the highest traffic, decreases with increasing traffic. In Fig. 3b the interfering node with a $1\,\mathrm{s}$ period pulse is introduced. We notice that in this situation CSMA is more sensitive than RAMAC to the increase of traffic in the network. Indeed, while remaining stable for low traffics, starting from medium traffic the CSMA's PDR drops quickly. At the same time, the RAMAC's PDR curve decreases linearly and gets closer to CSMA's one as the traffic increases. Finally, RAMAC shows about 2% higher PDR for $\lambda = 2.50$, meaning that it is less influenced by the interference. When the period of the interference node is $0.5\,\mathrm{s}$ (Fig. 3c), the two PDR curves become even closer with a maximum advantage of CSMA over RAMAC of about 3%. Similarly, we notice a better performance of about 3% of RAMAC in the case of $\lambda = 2.50$. Finally, in Fig. 3d we showcase a scenario in which an interfering node with a pulse with a $0.25\,\mathrm{s}$ period is present. Here we see that RAMAC has a higher PDR in all cases, proving that it is the best protocol to choose when the channel is noticeably affected by interference.

There are two reasons why the PDR performance of CSMA drops with the introduction of a disturbance while the performance of RAMAC is less influenced by the disturbance. The first is that while listening to the channel CSMA nodes have a high probability to hear a pulse coming from the disturbing node and in that case, they have to start the listening period again. The second is that the CSMA nodes use the frequencies also used by the disturbing node for all its packets (i.e., data, ACK) that can consequently collide with and be corrupted by the interfering signal. On the contrary, RAMAC recognizes the interfering pulses and does not use the frequencies where the interference is heard. In this way, even though RAMAC sacrifices some data rate capabilities with the control packets, it can keep the normal operations ongoing.

2) End-to-end latency: Fig. 4 depicts the end-to-end latency of the two protocols. Since CSMA is a protocol that listens to the channel every time before sending a packet, its end-to-end latency is expected to be higher than the one of RAMAC. When the disturbance is not present (Fig. 4a), we can see that the end-to-end latency of the two protocols is always of the same order of magnitude, with CSMA's latency being higher by about $0.5\,\mathrm{s}$ on average. However, the higher CSMA traffics suffer as soon as the disturbance is introduced, and with the increase of the frequency of its pulse. With interference with a period of $1\,\mathrm{s}$ (Fig. 4b), the end-to-end latency of CSMA for $\lambda = 1.87$ traffic is one order of magnitude greater than
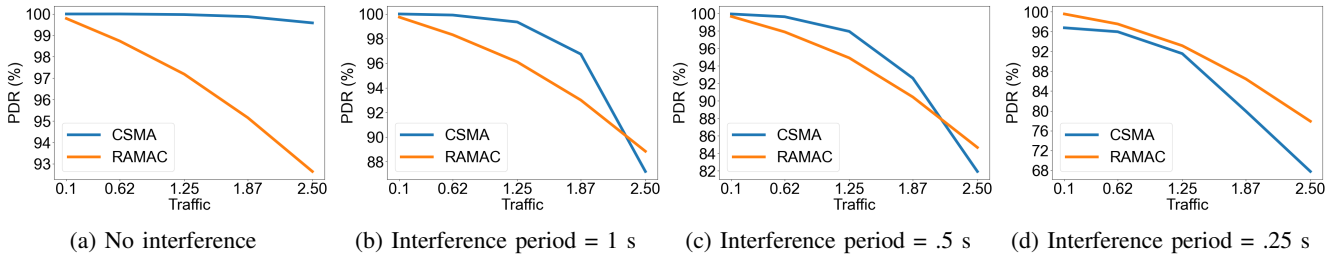
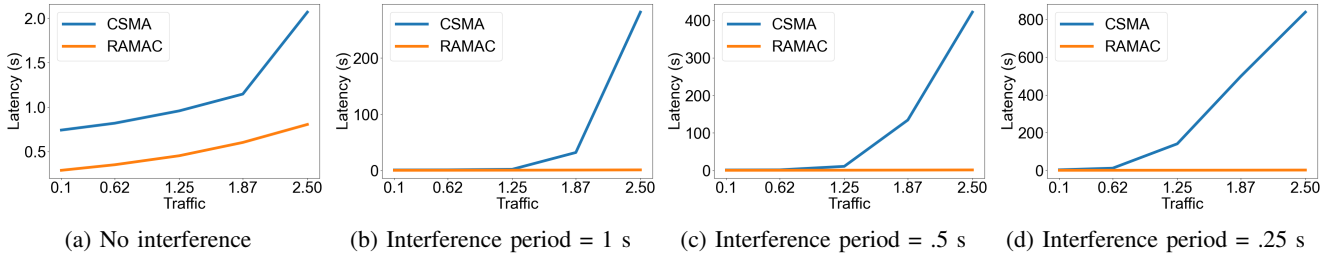Fig. 3: Packet Delivery Ratio; Packet Size 384 B.

Fig. 4: End-to-end latency; Packet Size 384 B.

the one of RAMAC and it's two orders of magnitude greater for $\lambda = 2.50$ traffic. When the interference period is 0.5 s (Fig. 4c), the end-to-end latency of CSMA is 1, 2, and 2.3 orders of magnitude greater than the RAMAC one for traffics with $\lambda = \{1.25, 1.87, 2.50\}$, respectively. For the highest traffic, CSMA's end-to-end latency reaches around 500 s, while RAMAC's end-to-end latency has an upper bound of 1.5 s. Finally, for interference with 0.25 s period pulse (Fig. 4d), the pattern showing the exponential growth of CSMA's end-to-end latency with the growing traffic is repeated. When $\lambda = 0.62$, the end-to-end latency of CSMA is already one order of magnitude bigger than RAMAC's end-to-end latency, and when $\lambda = 1.25$ it is two orders of magnitude greater than RAMAC's end-to-end latency. Moreover, CSMA's end-to-end latency keeps growing exponentially to arrive at almost three orders of magnitude greater than RAMAC's end-to-end latency when $\lambda = 2.50$. In this scenario RAMAC's end-to-end latency upper bound is 2 s. Overall, while the end-to-end latency of CSMA increases dramatically for higher traffic, reaching 840 s when an interference node is present, and making it unsuitable for real-time data delivery, the end-to-end latency of RAMAC, while still increasing, remains below 2 s.

E. Performance Results 1536 Byte Packets

Results from simulations in scenarios with packets of 1536 Bytes are shown in Fig. 5 and Fig. 6.

1) Packet delivery ratio: Results concerning the Packet Delivery Ratio can be seen in Fig. 5. The results follow the trends already seen for scenarios with 384 B packets: the performance of both protocols decreases when the traffic or interference increases. However, in this case, CSMA's PDR performance drops quicker than in the previous scenario since bigger data packets have a higher probability to collide with the interfering packets. In Fig 5a, the interference is absent but CSMA's

PDR drops with the increased traffic, and, for high traffic, achieves the same value as RAMAC. When the interference is introduced, we notice that only for the low traffic CSMA outperforms RAMAC, then it quickly drops to PDR values about 10% and 20% lower than the one achieved by RAMAC for interference frequencies of 1 s (Fig. 5b) and 0.5 s (Fig. 5c), respectively. Finally, when the frequency of the interference pulse is 0.25 s (Fig. 5d), RAMAC outperforms CSMA for all classes of traffic, obtaining up to $\sim 30\%$ better results. These trends are expected: with 1536 B packets, the higher data rate that the OFDM-based network can achieve offers a great advantage, moreover RAMAC, using the reservation mechanism and understanding and handling the presence of an interfering node, is able to minimize the number of collisions involving data packets.

2) End-to-end latency: Fig. 6 shows the end-to-end latency of the two protocols. As for the PDR, also the end-to-end latency in networks with packets of 1536 B has a behavior similar to the one with packets of 384 B, but with higher values. When the interference is absent (Fig. 6a) the end-to-end latency is comparable between the two protocols for low and medium traffic, but CSMA's latency is two orders of magnitude greater for higher traffics ($\lambda = \{1.87, 2.50\}$). When the interference with a pulse frequency of 1 s (Fig. 6b) and 0.5 s (Fig. 6c) is introduced, the end-to-end latency of CSMA at medium traffic ($\lambda = 1.25$) is two orders of magnitude greater than the end-to-end latency of RAMAC and becomes three orders of magnitude greater for high traffics. When the interference has a pulse frequency of 0.25 s (Fig. 6d), the end-to-end latency of CSMA is visibly higher than the one of RAMAC even for low traffic ($\lambda = 0.62$), and as in the previous case, it becomes three orders of magnitude bigger than RAMAC's end-to-end latency when the traffic is medium
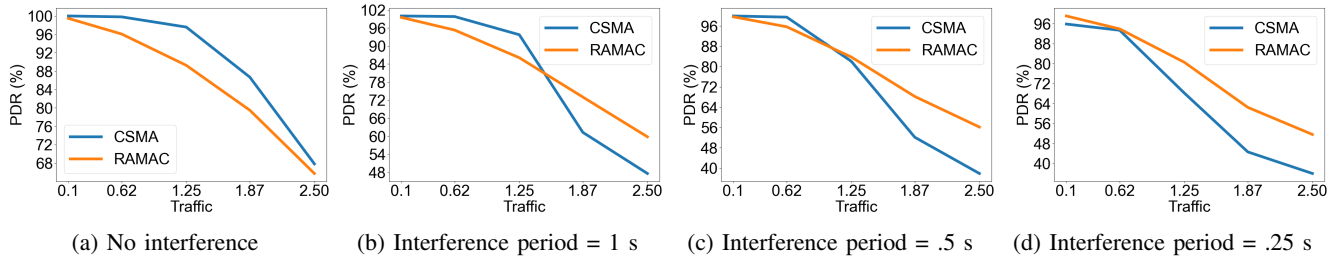
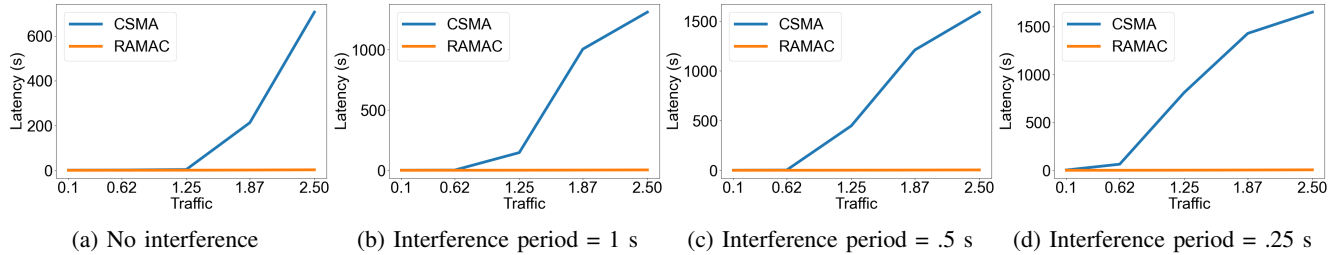Fig. 5: Packet Delivery Ratio; Packet Size 1536 B.

Fig. 6: End-to-end latency; Packet Size 1536 B.

and high. Again, while CSMA's end-to-end latency reaches values of about half an hour on average, RAMAC end-to-end latency has an upper bound of 6 s for all traffic classes, which makes it much more suitable for real-time applications.

V. CONCLUSIONS

The paper explores using OFDM-based physical layers underwater by defining a new OFDM-based MAC protocol, called RAMAC, for underwater networks. RAMAC leverages OFDM to avoid busy frequencies and a channel reservation mechanism to minimize data packet collisions. Given the higher spectral efficiency of OFDM-based systems, RAMAC is particularly useful when the network is busy, be it because of high traffic or large packets, and when the channel experiences selective interference. The performance of RAMAC and the possibilities afforded by OFDM-based systems are investigated by simulations in multi-hop scenarios with varying traffic, packet size, and interference frequency. Results show that RAMAC obtains PDRs that are comparable to or outperforms those of previous single-carrier solutions while maintaining an end-to-end latency lower than 6 seconds in all scenarios.

ACKNOWLEDGMENTS

REFERENCES

[1] S. A. H. Mohsan, A. Mazinani, N. Q. H. Othman, and H. Amjad, "Towards the Internet of Underwater Things: A comprehensive survey," *Earth Science Informatics*, vol. 15, pp. 735–764, 2022.

[2] H. Rohling, Ed., *OFDM: Concepts for Future Communication Systems*, ser. Signals and Communication Technology. Berlin Heidelberg, Germany: Springer-Verlag, March 23 2011.

[3] S. Zhou and Z. Wang, Eds., *OFDM for Underwater Acoustic Communications*. Hoboken, NJ: John Wiley & Sons, Ltd, March 28 2014.

[4] Y. Zhang, H. Wang, X. Shen, and R. Zhao, "An OFDM based channel aware access protocol for underwater acoustic networks of submarine pipeline monitoring," in *Proceedings of IEEE/MTS Global OCEANS 2015*, Washington, DC, October 19–22 2015, pp. 1–7.

[5] Z. Zhou, S. Le, and J.-H. Cui, "An OFDM based MAC protocol for underwater acoustic networks," in *Proceedings of ACM WUWNet 2010*, Whoods Hole, MA, September 30–October 1 2010, pp. 1–8.

[6] Y. Yin, S. Roy, and P. Arabshahi, "A modified CSMA/CA protocol for OFDM underwater networks: Cross layer design," in *Proceedings of ACM WUWNet 2015*, Washington, DC, October 22–24 2015, pp. 1–8.

[7] Y. Su, Y. Zuo, Z. Jin, and X. Fu, "OSPG-MAC: An OFDMA-based subcarrier pregrouping MAC protocol for underwater acoustic wireless sensor networks," in *Journal of Sensors*, vol. 7, June 2019, pp. 1–12.

[8] E. Demirors, J. Shi, A. Duong, N. Dave, R. Guida, B. Herrera, F. Pop, G. Chen, C. Cassella, S. Tadayon, M. Rinaldi, S. Basagni, M. Stojanovic, and T. Melodia, "The SEANet project: Toward a programmable Internet of Underwater Things," in *Proceedings of IEEE UComms 2018*, Lerici, Italy, August 28–30 2018, pp. 1–5.

[9] S. Falleni, D. Unal, A. Neerman, K. Enhos, E. Demirors, S. Basagni, and T. Melodia, "Design, development, and testing of a smart buoy for underwater testbeds in shallow waters," in *Proceedings of IEEE/MTS Global OCEANS 2020*, Singapore–U.S. Gulf Coast, October 5–14 2020, pp. 1–7.

[10] F. Campagnaro, R. Francescon, F. Guerra, F. Favaro, P. Casari, R. Diamant, and M. Zorzi, "The DESERT underwater framework v2: Improved capabilities and extension tools," in *Proceedings of UComms 2016*, Lerici, Italy, August 30–September 1 2016, pp. 1–5.

[11] "DESERT Underwater." [Online]. Available: http://desert-underwater.dei.unipd.it

[12] S. Falleni, D. Saha, I. Haque, T. Melodia, and S. Basagni, "Development and testing of an OFDM physical layer for the DESERT simulator," in *Proceedings of IEEE/MTS Global OCEANS 2021*, San Diego–Porto, September 20–23 2021, pp. 1–7.

[13] D. Unal, S. Falleni, E. Demirors, K. Enhos, S. Basagni, and T. Melodia, "Software-defined underwater acoustic networking platform for underwater vehicles," in *Proceedings of IEEE ICC 2022*, Seoul, South Korea, May 16–20 2022, pp. 1–6.

[14] W. Shieh and D. I., Eds., *Orthogonal Frequency Division Multiplexing for Optical Communications*. Elsevier, 2009.