Field Tests of the Software Defined Modem Prototype for the MODA Project

Emanuele Coccolo, Roberto Francescon, Filippo Campagnaro, Michele Zorzi

***Abstract*—Underwater acoustic communications has developed to such a stage that new applications and functionalities emerged in various fields of the human activities, from military missions to commercial operations. A new field of application is the use of underwater vehicles in swarms, operating in a coordinated fashion to perform tasks that would be harder, or dangerous, for humans or single vehicles. Swarms of underwater drones require the use of very flexible and high-performance wireless modems: in the context of a national funded project we developed a Software Defined Modem (SDM) to be employed in lightweight underwater vehicles. This work presents the tests of the communications capability of the modem in a very shallow environment. The tests proved the validity of the modem design, still at the prototype stage, and shed light on the areas that are worth investigating to further improve its performance. Additionally, we demonstrate that it is possible to build an SDM, capable of real-time communications, using only off-the-shelf and easily accessible hardware and software tools.**

***Index Terms*—Underwater acoustic modem, software defined modem, sea-trial, tests, low-cost underwater modem.**

I. Introduction

Underwater wireless communications has witnessed a rapid development during the last 20 years, both in the research field and at the commercial level [1]–[4]. In particular, acoustic underwater communications improvements allowed to reach a performance baseline for off-the-shelf devices, that gave rise to new applications and functionalities. Many human activity fields, nowadays, rely on the performance of underwater communications devices, especially acoustic ones, to carry out their tasks: from military and control operations to scientific and oceanic research, oil and gas facilities maintenance and offshore wind farms operations [5]. Lately, the use of Autonomous Underwater Vehicles (AUVs), grouped together in fleets and working in a coordinated fashion, demonstrated the ability to perform even more advanced tasks and posed new challenges to underwater acoustic communications [6]–[8]: compact, lightweight and energy-efficient acoustic modems are required to be installed on size-limited, battery-powered vehicles. These challenges could be partially overcome by employing Software Defined Modems (SDMs): their flexibility and reconfigurability allow to tailor many characteristics of the communication stack to the specific channel environment, thereby improving reliability, lowering energy consumption and limiting the hardware to few processing units [9]. In the last decade SDMs have proven to be a technology able to tackle these challenges: their adaptability allows to efficiently use the limited resources of the underwater node, while the general purpose processing units they run on are much less expensive then specialized hardware, such as digital signal processors or field programmable gate array circuits.

Our research group designed and developed a software defined modem in the context of an Italian national funded project, in collaboration with the Italian General Directorate of Naval Armaments (NAVARM) and the Italian National Research Council (CNR): the MODA project. This modem includes 2 layers of the communication stack, namely the physical (PHY) layer and the Medium Access Control (MAC) layer. The distinctive feature of this modem will be the ability to perform One Way Travel Time (OWTT) range estimation, relying on high precision clocks, together with data transmission [10]. In this work we are going to present the results of the first field tests of the data transmission functionality of the modem, illustrating the main challenges when designing a software defined modem with advanced capabilities. After only one year of development, the modem is able to perform all operations in real-time, without the need of any post-processing. The main novelty of this work is the creation of an experimentation SDM platform that uses only low-cost off-the-shelf software and hardware tools, easy to be integrated by researchers, students and practitioners, bringing underwater communications closer to civilian applications and enabling dense network deployments. Nevertheless, using high performance transducers and atomic clocks also large scale off-shore applications can be supported by the same platform, thus the system can be quickly adapted to different application requirements.

In Section II the various parts that make up the modem are described, along with the setup used in the field trials. Section III illustrates the tests (how we performed the evaluation and the choices we made), while Section IV presents the results. Finally, Section V draws our concluding remarks on the performance reached and discusses future developments of the modem.

II. Tests setup

The setup employed in the tests was composed by two Software Defined Modems (SDM) developed by our research group: one of the modems was transmitting and the other was receiving. The modems were powered by Ni-Cd batteries: the receiver was powered at 12 V, while the transmitter was powered at both 24 V and 48 V to test different configurations.

F. Campagnaro (Corresponding author, email: campagn1@dei.unipd.it), E. Coccolo and M. Zorzi are with the Department of Information Engineering, University of Padova, Italy. R. Francescon and M. Zorzi are with Wireless and More srl, Padova, Italy.

This work has been partially supported by the Italian General Directorate of Naval Armaments (NAVARM) and the Italian National Plan for Military Research (PNRM) (contract 20593), and by the European Union - FSE REACT EU, PON Research and Innovation 2014-2020 (DM 1062/2021).

Fig. 1. Tests setup: box containing the boards, feeding batteries (left, red-colored) and control laptop (right)

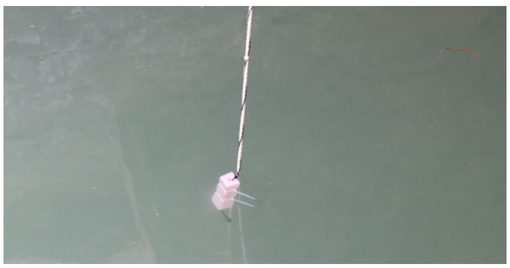

Fig. 2. Acoustic transducer lowered into the water

The modems were running on a Raspberry Pi 4 Single Board Computer (SBC) together with a *HiFiBerry* high fidelity audio module [11], and were accessed via SSH connection from a laptop. Some ropes and weights allowed to deploy the transducers at selected depths and keep them in place. The hardware components, power supply batteries and the laptop used for managing the modems can be seen in Fig. 1

One unit of a complete modem is composed of four main components: a processing unit, Raspberry Pi 4 Model B mounted with a high performance DAC/ADC (the HiFiBerry module); a transmission amplifier, specifically an AB type to avoid frequency cutoff [12]; a pre-amplifier for the received signal; two hydrophones AS-1 [13] to physically transmit the signal. Further information about the hardware setup is provided in [10], where all the components are described in detail. While the SBC, along with the amplifiers and the power supply batteries, were standing on a bridge and on a jetty, the acoustic transducers, safely kept in place by some weights, were drawn down into the water along the ropes.

The tests were performed in the Piovego river passing through the city of Padua (Italy) which had a depth, at the time of the tests, of 1.8 m at the center: the scenario is then that of a very shallow water acoustic communication. More details are provided in Section III.

III. Description of the Tests

The tests were performed by sending packets with fixed length from the transmitter to the receiver: the packets sizes were 32 bytes and 5 bytes at the application level. The subsequent levels, MAC and PHY, add 26 bits and 8 bytes,

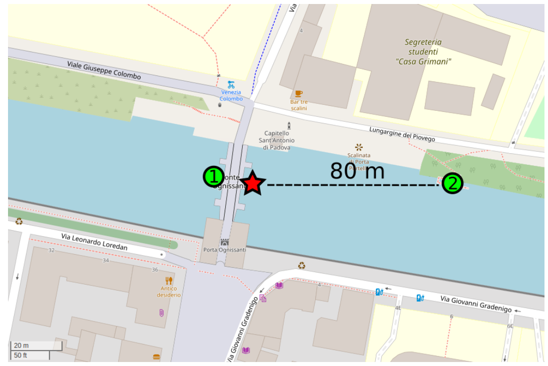

Fig. 3. Test location. The red star represents the position of the transmitter, fixed in both tests; the two green circles are the locations of the receiver, the first at 7 m and the second at 80 m.

respectively, with the basic information for ranging operations and demodulation of the payload.

For this kind of test, we limited the amount of components per single apparatus: there were only two devices, one being only a receiver and the other only a transmitter. In this way, packet transmissions were performed only by one node which allowed us to identify possible errors introduced by the channel and the configuration of the environment, with an analysis of the signal at the receiver side.

Description of the location. The tests were performed in the Piovego River, a small branch of bigger rivers passing near the city of Padua, Italy, and characterized by a very weak current and a shallow riverbed. Indeed, in the location of the test (45.40989, 11.89276), the center of the river was excavated to permit the passage of small vessels. Moreover, the bottom is composed of mud, which does not scatter excessively the acoustic waves, and has a depth of 1.80 m in the central part of the canal, and 0.3 m on the sides.

To evaluate the attenuation of the channel, we identified two topologies with different distances between the nodes. The first consists of the two nodes placed at opposite sides of the Portello Bridge, which has a width of 7 m; in the second, the transmitter is still placed on the bridge and the receiver node is on a service jetty at approximately 80 m. Furthermore, there are two wooden structures, used as a channel marker (typical in the Venetian Lagoon and connected canals), in the proximity of the testing site, but they do not compromise the LoS (Line-of-Sight) between the modems. The topology setup for both these cases can be seen in Fig. 3.

System settings. The software defined modem has been developed with a modular approach, consisting mainly of a physical layer (PHY) and a data link layer, with both LLC (Logical Link Control) and MAC (Medium Access Control) features. The PHY module takes advantage of two other modules to perform the signal generation, detection and decoding, and to transmit or receive the acoustic waves. For the first module, we used a wrapper of `flexframe`, provided in the *liquid-dsp* library [14], which allowed us to easily switch between different modulations and Forward Error Correction (FEC) schemes. As an extension to the existing frame generator and synchronizer, we attached an interpolator, coupled with a specular decimator at the receiver, to obtain a higher amount of audio samples so that the symbol duration becomes longer in the time domain. Instead, for the data acquisition module, we opted for a wrapper of the ALSA library [15], which permits to use the audio pipeline integrated in the operating system; this system introduces a certain amount of latency, which however is constant with respect to the buffer size, and thus can be evaluated and subtracted as a constant delay due to processing operations.

Regarding the settings of the system, which cover mainly the Digital Signal Processing (DSP) and transmission protocols, we decided to perform the tests changing just a few combinations of parameters, limited to the DSP: the MAC policy remained fixed, since only one node was transmitting, thus every 7 s the transmitter sent a packet containing a progressively generated string (`S000, S001, S002, ... S100` for the 5 bytes packet, and a similar one for the 32 bytes packet). The interval period was chosen before the tests, examining the amount of time required for a single packet interpolated with the highest number of samples per symbol to be transmitted entirely.

The parameters taken into account during this trial were the samples per symbol (SPS) used to represent each symbol generated by the *liquid-dsp* frame generator, the modulation scheme, the inner and outer FEC, the voltage for powering the transmission amplifier (either 24 V or 48 V). Other noteworthy parameters are the *buffer time* and *period time* of the ALSA library, which represent the total amount of time that can be contained in the ALSA ring buffer, and the interval between two system interrupts that activates the buffer filling. If set too low, the audio appears crackling and there is a significant quality loss; if set too high, the latency of the ALSA library increases proportionally. In our case, they were 40'000 μs and 400 μs; these depend on the system where the SDM is running and specifically on the audio card capabilities.

The default Forward Error Correction (FEC) used for the payload was Hamming 7/4. Then, after a first iteration of the test, the FEC was changed using the same combination provided in the *liquid-dsp* header, which is the inner FEC with SECDED7264 and the outer FEC with Hamming 8/4. Regarding the modulation, the first part of the tests was performed with BPSK with carrier frequency $f_c = 50$ kHz and bandwidth $f_c/4$ and, in a second part, it was shifted to QPSK but with very poor results. For the samples per symbol in the interpolator setting, the values we chose were 10, 20, 50, and given that the frame generator produces 2 samples per symbol, the resulting final number of samples is doubled.

IV. Results of the tests and performance analysis

First, we recorded the reception of one packet of 32 bytes to determine, visually, if the packet was received with enough power to be at least detected by the frame synchronizer. The spectrogram can be seen in Fig. 4. Here, the right part has been selected to outline the multipath introduced by the channel, whose length is approximately 30 ms.

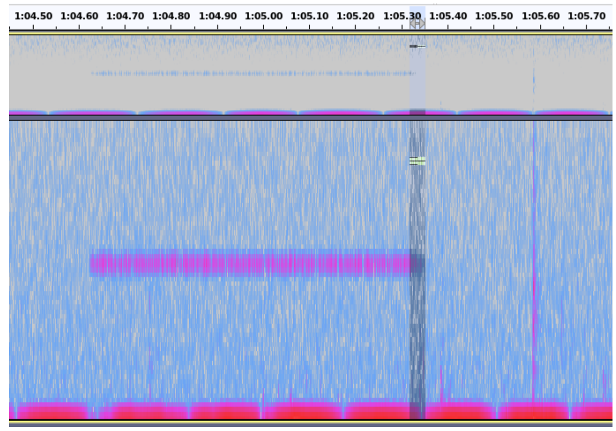

Fig. 4. Spectogram of a 32 bytes packet, obtained with Audacity. The selected part at the right end of the frame is the multipath contribution of the received packet (approximately 30 ms).

Our analysis is based on the correct receptions of the packets transmitted: this allows us to retrieve the Packet Delivery Ratio (PDR) over a period of time. In Tabs. I and II, the PDR of 101 packets transmitted between the two sides of the bridge are shown, in particular we note that the amplifier at the transmitter was powered with 24 V and we did not perform the tests at 48 V because the multipath was excessive due to the shallow scenario, to the point where there was a decay in performance.

TABLE I
PDR results of 101 packets of 5 bytes, performed at a distance of 7 m with transmitter and receiver at the sides of a bridge. FEC0 is the inner FEC and FEC1 is the outer FEC.

PDR	10 sps	20 sps	50 sps
default FEC: fec0 = Hamming 7/4 fec1 = None	0.90	0.96	0.81
FEC as header: fec0 =SECDED7264 fec1 = Hamming 8/4	0.60	0.73	-

TABLE II
PDR results of 101 packets of 32 bytes, performed at a distance of 7 m with transmitter and receiver at the sides of a bridge. FEC0 is the inner FEC and FEC1 is the outer FEC.

PDR	10 sps	20 sps	50 sps
default FEC: fec0 = Hamming 7/4 fec1 = None	0.63	0.72	0.69
FEC as header: fec0 =SECDED7264 fec1 = Hamming 8/4	0.79	0.67	-

In Figs. 5 and 6, we provide a summary plot with the results of 5-byte and 32-byte packets transmitted from the bridge to the nearby service jetty (distance of 80 m). During these tests, besides modifying the voltage for the transmission amplifier, we changed the depth of the receiver hydrophone. This led to very different results, which means that in this scenario the geometry of the network is a key factor, especially for how the sound waves are reflected in each position.

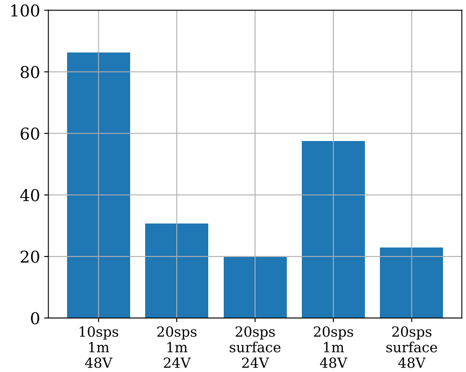

Fig. 5. Summary plot of PDR obtained sending 101 packets with length 5 bytes between the bridge and the service jetty (distance 80 m). On the x-axis, it is specified if the test was either with the receiver at 1 m of depth or just below the surface, and the voltage of the amplifier is mentioned as well.

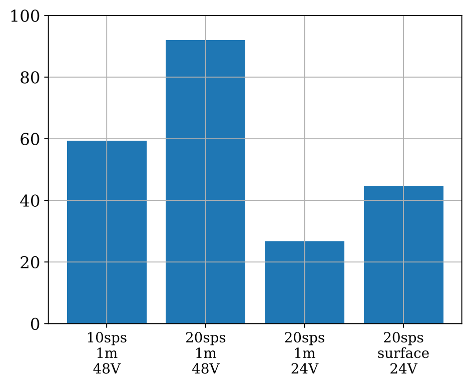

Fig. 6. Summary plot of PDR obtained sending 101 packets with length 32 bytes between the bridge and the service jetty (distance 80 m). On the x-axis, it is specified if the test was either with the receiver at 1 m of depth or just below the surface, and the voltage of the amplifier is mentioned as well.

These results highlight two consequences: at short distance, increasing the symbol length brings little to no improvements as the packets were successfully decoded even with just 10 samples per symbol; at longer distances, bigger lengths definitely increase the robustness of the packet, in conjunction with a higher transmission power. Secondly, using a concatenated FEC at short distance does not improve the performance of the modem and a simple Hamming 7/4 is more than sufficient to obtain a valid link between the nodes. This leads to an analysis of the behavior of the *liquid-dsp* frame synchronizer: first, it tries to detect a preamble to determine the beginning of a packet; when this happens, the known preamble is analyzed to determine the channel impairments, such as the carrier offset and the EVM (error vector magnitude). These parameters are later used during the decoding phase to adjust the decoding to the right symbols, but, since the underwater acoustic channel is time-variant, they can be different during the transmission of a single packet if this is long enough. This is the reason why longer packets were not more robust at shorter distances and a stable link could not be establish with 50 samples per symbol at a distance of 80 m. Moreover, the problem of having a packet too long, to be sent through the channel, is as well connected with the issue of using a concatenated FEC, since the number of bits that are resulting from this encoding is far greater.

Furthermore, these tests showed us that high frequencies can be successfully used in very shallow water (between 1.2 m and 1.8 m max) using a shift-keying modulation. For this project, we performed an estimation of the bitrate at the physical layer and, considering FECs and additional headers, the value is 1.3 kbit/s, using a 32-byte packet at 10 samples per symbol for interpolation. Naturally, with higher values of samples per symbol used in the interpolator, the bitrate decreases proportionally, a little less than linearly due to an increase of the processing time between packets.

Regarding the QPSK modulation scheme, a viable link could not be established between the nodes at a 7 m distance, as out of 100 packets less than 10 were received correctly in several iterations of the experiment. For this reason, the results were omitted in this work and will be taken into consideration in the future, when new FEC schemes will be introduced, along with an improvement of the transmission amplifier.

We finally performed preliminary tests involving the OWTT capability of the modem: by inserting a timestamp into the packets, the receiver was allowed to perform an estimate of the time of flight (ToF) of the packet received. These tests resulted in a mean value of 5 ms and 54 ms for, respectively, the 7 m and the 80 m range tests. Considering a sound speed of 1480 m/s for the freshwater of the canal, we obtain estimated distances of 7.4 m and 79.92 m, showing an extremely high accuracy despite the challenging propagation conditions caused by the shallow water environment.

V. Conclusion

In this paper, we showed the tests performed to evaluate a Software Defined Modem for the MODA project. These tests were performed nearby the Department of Information Engineering of the University of Padua, specifically in the Piovego river. This location is characterized as a very shallow canal with a center zone excavated to permit the travelling of lightweight vessels. The tests demonstrated the successful initial design of the modem: the two modems were able to transmit and receive with performance in line with previous and current works [16], [17]. The promising results of these tests support the current design, especially if we consider the very challenging environment in which we were operating. The tests also showed that having a packet with a long transmission time can worsen the robustness of the communication, since the channel parameters can often change from the preamble until the end of the packet.

As future developments, new modulation and coding schemes will be added and tested, such as spread spectrum modulations, along with other more efficient FEC algorithms, such as convolutional codes and Reed-Solomon codes. The ranging capability through OWTT estimation will be tested as well, using both OCXO Time server and boards with atomic clocks.

REFERENCES

[1] E. Sozer, M. Stojanovic, and J. Proakis, "Underwater acoustic networks," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 72–83, Jan. 2000.

[2] D. Pompili and I. F. Akyildiz, "Overview of networking protocols for underwater wireless communications," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 97–102, Jan. 2009.

[3] K.-H. Park, M. A. Imran, P. Casari, H. Kulhandjian, H. Chen, A. Abdi, and F. Dalgleish, "IEEE Access Special Section Editorial: Underwater wireless communications and networking," *IEEE Access*, vol. 7, pp. 52 288–52 294, Apr. 2019.

[4] M. F. Ali, D. N. K. Jayakody, Y. A. Chursin, S. Affes, and S. Dmitry, "Recent advances and future directions on underwater wireless communications," *Archives of Computational Methods in Engineering*, vol. 27, no. 5, pp. 1379–1412, Nov. 2020.

[5] A. Song, M. Stojanovic, and M. Chitre, "Editorial underwater acoustic communications: Where we stand and what is next?" *IEEE Journal of Oceanic Engineering*, vol. 44, no. 1, Jan. 2019.

[6] C. Lin, G. Han, M. Guizani, Y. Bi, J. Du, and L. Shu, "An SDN architecture for AUV-based underwater wireless networks to enable cooperative underwater search," *IEEE Wireless Communications*, vol. 27, no. 3, pp. 132–139, Jun. 2020.

[7] Y. Yang, Y. Xiao, and T. Li, "A Survey of Autonomous Underwater Vehicle Formation: Performance, Formation Control, and Communication Capability," *IEEE Communications Surveys Tutorials*, vol. 23, no. 2, pp. 815–841, Secondquarter 2021.

[8] X. Zhuo, M. Liu, Y. Wei, G. Yu, F. Qu, and R. Sun, "AUV-Aided Energy-Efficient Data Collection in Underwater Acoustic Sensor Networks," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10 010–10 022, Oct. 2020.

[9] H. S. Dol, P. Casari, T. van der Zwan, and R. Otnes, "Software-Defined Underwater Acoustic Modems: Historical Review and the NILUS Approach," *IEEE Journal of Oceanic Engineering*, vol. 42, no. 3, pp. 722–737, Jul. 2017.

[10] E. Coccolo, F. Campagnaro, D. Tronchin, A. Montanari, R. Francescon, L. Vangelista, and M. Zorzi, "Underwater Acoustic Modem for a MOrphing Distributed Autonomous Underwater Vehicle (MODA)," in *Proc. MTS/IEEE Oceans*, Chennai, India, Feb. 2022.

[11] ""HiFiBerry"," Last time accessed: July 2022. [Online]. Available: https://www.hifiberry.com/

[12] "LM3886 reference sheet," Last time accessed: July 2022. [Online]. Available: https://www.ti.com/product/LM3886

[13] ""AS-1 Hydrophone"," Last time accessed: July 2022. [Online]. Available: https://www.aquarianaudio.com/as-1-hydrophone.html

[14] "`liquid-dsp` library," Last time accessed: July 2022. [Online]. Available: https://liquidsdr.org/

[15] "ALSA project - the C library reference," Last time accessed: July 2022. [Online]. Available: https://www.alsa-project.org/alsa-doc/alsa-lib/index.html

[16] L. Wu, J. Trezzo, D. Mirza, P. Roberts, J. Jaffe, Y. Wang, and R. Kastner, "Designing an adaptive acoustic modem for underwater sensor networks," *IEEE Embedded Systems Letters*, vol. 4, no. 1, pp. 1–4, Dec. 2012.

[17] E. Demirors, G. Sklivanitis, G. E. Santagati, T. Melodia, and S. N. Batalama, "A high-rate software-defined underwater acoustic modem with real-time adaptation capabilities," *IEEE Access*, vol. 6, pp. 18 602–18 615, Mar. 2018.