

# ahoi: Inexpensive, Low-power Communication and Localization for Underwater Sensor Networks and $\mu$ AUVs

BERND-CHRISTIAN RENNER, JAN HEITMANN, and FABIAN STEINMETZ,  
Hamburg University of Technology, Germany

The recent development of small, cheap AUVs enables a plethora of underwater near- and inshore applications. Among these are monitoring of wind parks, detection of pollution sources, water-quality inspection, and the support of divers during disaster management. These tasks profit from online reporting, control, and AUV swarm interaction; yet they require underwater communication. Unfortunately, commercial devices are prohibitively expensive and typically closed-source, hampering their application in affordable products and research. Therefore, we developed the open-source<sup>1</sup> *ahoi* acoustic modem. It is (i) small enough to be carried by micro AUVs, (ii) consumes little enough energy to not diminish operation times of its host, (iii) comes at an attractive unit cost below \$600, (iv) can reliably communicate at distances of 150 m and more, and (v) supports ranging without additional hardware. Due to its modular build, the modem can be customized and is suitable as research platform to analyze, e.g., MAC and routing protocols. We conducted extensive real-world studies and present results of communication range, packet reception rate, ranging accuracy, and efficient and reliable self-localization. Finally, we draw conclusions regarding acoustic communication, ranging, and localization with inexpensive and low-power devices that go beyond a particular device. Our study, hence, encompasses general insights, observations, and recommendations.

CCS Concepts: • Networks → Network experimentation; • Computer systems organization → Embedded hardware; Robotic components; Firmware; • Hardware → Wireless devices;

Additional Key Words and Phrases: Acoustic, underwater, communication, localization, AUV, swarm, ahoi, modem

## ACM Reference format:

Bernd-Christian Renner, Jan Heitmann, and Fabian Steinmetz. 2020. ahoi: Inexpensive, Low-power Communication and Localization for Underwater Sensor Networks and  $\mu$ AUVs. *ACM Trans. Sen. Netw.* 16, 2, Article 18 (January 2020), 46 pages.

<https://doi.org/10.1145/3376921>

18

<sup>1</sup>see [www.ahoi-modem.de](http://www.ahoi-modem.de).

This work has been partially supported by the German Federal Ministry of Education and Research (BMBF) under grant number 13N14153, the German Federal Ministry for Economic Affairs and Energy (BMWi) under grant number 03SX463C, and the ERA-NET Cofund MarTERA (contract 728053).

Authors' addresses: B.-C. Renner, J. Heitmann, and F. Steinmetz, Hamburg University of Technology, Am Schwarzenberg-Campus 1, Hamburg, Hamburg, 21073, Germany; emails: christian.renner@tuhh.de, jan@heitmann.it, fabian.steinmetz@tuhh.de.



This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2020 Copyright held by the owner/author(s).

1550-4859/2020/01-ART18

<https://doi.org/10.1145/3376921>

## 1 INTRODUCTION

Wireless networked sensing has already conquered many application domains over the past two decades and is currently extending to the field of inshore and coastal underwater monitoring and control. Starting from stationary underwater sensor networks [17, 32, 39, 45, 54, 55], a recent focus on co-operative, autonomous, mobile underwater robot swarms [20, 30, 36, 47, 60, 61] has evolved in both academia and industry. Novel miniature mobile underwater robots with lengths well below 1 m and a unit cost between some hundred and a few thousand dollars have become available [30, 34, 48, 59]. Due to their size and low cost, these robots are frequently called micro autonomous underwater vehicles ( $\mu$ AUV). They are typically equipped with an inexpensive inertial measurement unit (IMU) and cheap, application-specific sensors. Figure 1 shows the MONSUN  $\mu$ AUV with available sensors.

These  $\mu$ AUVs ultimately enable automated, unsupervised environmental underwater near- and inshore monitoring, inspections, and interaction with the environment. Applications range from automated water-quality monitoring over inspections of bridges and wind parks to supporting divers in dangerous missions. Specific needs we identified when interviewing local administration in Hamburg for the MoSAIk [57] research project are:

- the timely detection of health hazards for the public,
- bio-hazards caused by fuel-oil leaks and other pollutant introductions by ships or industrial equipment,
- examination of ship hulls and sheet piles, and
- disaster management support.

These tasks typically require fast reaction times and massively profit from collaboration of multiple robots. Figure 2 illustrates a possible scenario, in which two  $\mu$ AUVs perform an underwater monitoring task and report their measurements via a relay station at the surface to a control center. The latter may send instructions back to the  $\mu$ AUVs. A great advantage of such cooperation is that robots performing an underwater task do not have to surface for communicating with the control center. While communication may be relayed by fixed buoys, swarm interaction is an alternative that relaxes the demand for infrastructure. However, both options have their right to exist and corresponding use case.

Either way, underwater communication is a mandatory requirement as enabling technology. Since radio waves suffer from heavy absorption in water, and poor visibility in many waters excludes optical approaches, only acoustic communication appears suitable. Unfortunately, we found that available acoustic modems [10, 16, 24, 26, 27, 50, 65, 70, 72, 86] cannot be easily integrated in  $\mu$ AUVs such as MONSUN [48] or HippoCampus [30] with tight constraints on size, battery capacity, modifiability, and cost (we elaborate on this in Section 2) or due to availability problems. Furthermore, real-world experiments involving acoustic communication—as, e.g., carried out in References [21, 52]—are essential for developing and assessing practical MAC and routing protocols as well as distributed underwater localization and swarm algorithms. However, there is a lack of practical experience regarding their use and performance in  $\mu$ AUVs. A recent, preliminary study has been presented in Reference [71], which analyzes communication quality of a custom acoustic modem in static and mobile cases, yet does not include acoustic ranging and mobile  $\mu$ AUV localization.

For the above purposes, relatively inexpensive, customizable, and easily reproducible acoustic modems are required, which are not yet available or are difficult to obtain. We therefore designed and built the *ahoi* acoustic modem for underwater communication that serves two purposes. First,

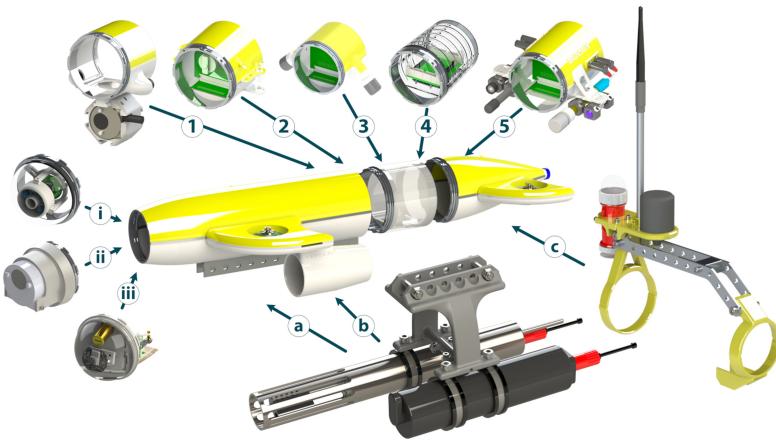


Fig. 1. MONSUN  $\mu$ AUV [48] with available sensor equipment including, e.g., an early version of the *ahoi* modem and the option to add a professional CTD sensor. The main body of the robot has a length of ca. 60 cm with a diameter of 10 cm. The cost of a fully equipped MONSUN (material plus electronics, without sensors) is ca. \$8K; weight is approximately 7 kg (in air).

it is tailored to the particular needs of  $\mu$ AUVs regarding size, price, and energy consumption. Second, it simplifies and enables real-world experiments in the domain of autonomous  $\mu$ AUV swarms and their communication for both industry and academia at an affordable price point, with readily available components and with the freedom to modify hardware and software components, as we intend to release the *ahoi* modem as open source. Being a software-defined device, the *ahoi* modem enables comparison and testing of, e.g., novel modulation schemes [66] on real hardware. With this device at hand, thorough analysis of real-world acoustic communication and localization, in static and mobile use cases, is enabled and conducted.

### 1.1 Contributions

We make the following contributions that are centered around the *ahoi* acoustic modem but yield, via extensive outdoor experiments, general findings for acoustic underwater communication, ranging, and localization in shallow waters.

- We present the *ahoi* modem, an inexpensive and low-power device, designed as research tool for practical evaluation of underwater communication and localization algorithms and protocols and tailored to the needs of  $\mu$ AUVs, matching their price and size demands.
- We discuss design decisions and the architecture of our modem, including both hard- and software aspects, with particular focus on modularity and extensibility in Section 3.
- We present the results of an extensive evaluation of the hardware in the laboratory to assess its consumption and quality of the analog circuitry in Section 4.
- We report on an extensive real-world experiment to showcase the performance of the *ahoi* modem in terms of communication range, packet reception rates, and ranging accuracy in the stationary case in Section 5. We also report on general findings and challenges of the acoustic channel in shallow waters.
- We integrated the *ahoi* modem in the HippoCampus  $\mu$ AUV and present the results of a mobile, outdoor measurement campaign, in which we analyzed communication quality—compared to the static case—and inspected the capability of acoustic underwater localization in Section 6. We also generalize our findings.

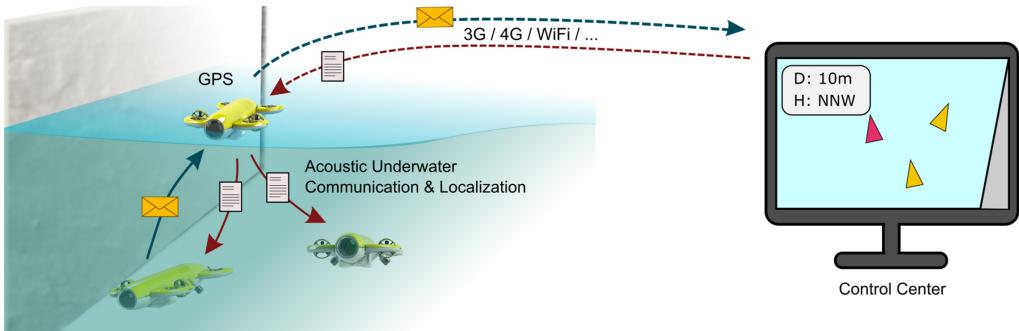


Fig. 2. Potential application scenario for underwater swarms with a mobile relay station at the surface to enable fast response times without the need for surfacing of each robot.

- Finally, we pinpoint general challenges of the acoustic channel and give pointers to follow-up research.

## 1.2 Open-source Release

All sources of the modem hardware, firmware, and tools are released under a permissive open-source license. A brief overview and links to the public repositories are found at [www.ahoi-modem.de](http://www.ahoi-modem.de).

## 2 BACKGROUND AND RELATED WORK

First, we revisit typical underwater application scenarios and position our modem to existing devices to support the need for its development. We provide the theoretical background of acoustic (underwater) communication and stress its peculiarities.

### 2.1 Applications

Exploration and monitoring of underwater sceneries are drawing considerable attention [25, 32]. Much research and industrial effort has been dedicated to deep-sea applications, e.g., Reference [58]; yet relatively little effort has been spent on shallow-water sceneries such as coastal and in-shore environmental monitoring and disaster management at depths of normally double digits or at most a few hundred meters. Among existing applications are the investigation of sub-mesoscale eddies with miniature underwater robots in Reference [47], ship tracking with an underwater sensor system in harbors [55], and a low-cost underwater sensor node called HydroNode [54]. Recently, the EU-funded research projects SUNRISE [53] and SUNSET [39] have contributed to the development of an Internet of Underwater Things (IoUT). Examples for inshore and shallow-water applications are the observation of drinking water [87] and water-quality [57]. Another use case that is expected to gain popularity and industry adoption is that of tank inspections [43].

Fine-grained measurements require a dense network of stationary sensors [45, 54], mobile devices—so-called micro autonomous underwater vehicles ( $\mu$ AUV) due to their small size [30, 34, 48, 59]—or an ecosystem of both.  $\mu$ AUVs may act on their own or collaborate in swarms [36]. In both cases,  $\mu$ AUVs rely on underwater communication and self-localization for navigation and to report readings timely to a control center. This enables the operator, e.g., a public authority, to take countermeasures in case of an identified pollution source or instruct the  $\mu$ AUVs to inspect and explore a conspicuous area in detail. Stationary underwater sensors also profit from wireless communication by reducing the amount of cables and cost while increasing installation flexibility.

Typically, acoustic communication is the method of choice, because it offers a range of several hundred meters and copes well with various water conditions. For details on the advantages of acoustic communication and a comparison with alternatives, the reader is referred to, e.g., Reference [29].

## 2.2 Acoustic Modems

In the past decade, several research groups and companies have designed and presented modems for acoustic underwater communication. We position our acoustic modem against the most prominent devices. A detailed comparison table in Appendix A gives a condensed overview, and a comprehensive and very recent study is also available in Reference [63].

Commercial modems are off-the-shelf solutions and, in theory, only need a few wires, manual reading, and (usually serial) interfacing with the robot. However, their unit cost likely overshoots that of the host by a multiple, hence rendering them unattractive for (swarm) research; e.g., the Evologics S2C M HS [24] comes at roughly \$10K per unit. Our modem comes at \$600 material cost plus assembly, which can be realized at very low costs at most research institutes. Other modems—e.g., Teledyne ATM-903 [72], Sonardyne Modem 6 Sub-Mini [65], the AppliCon SeaModem [16], and the developic HAM.BASE [19]—are too large for easy integration in  $\mu$ AUVs. Our modem has been tailored to fit into MONSUN and HippoCampus, one of the smallest available  $\mu$ AUVs. In contrast, the Aquacomm Gen2 [22] is too bulky for even larger  $\mu$ AUVs like HippoCampus. Moreover, all commercial solutions do not allow access to the firmware, hence rendering real-world experiments with, e.g., novel modulation or MAC protocols difficult, if not impossible. Most devices also restrict network size to 15 or 64 nodes and have high power consumption of 10 W to 100 W during transmission. The Tritech Micron [86] has an extremely low data rate of only 40 bit/s, likely hindering swarm communication. Our modem improves on the data rate by a factor of six or more, depending on configuration. With 100 bit/s, the SeaTrac X110 [11] is merely faster at a relatively high consumption of 500 mW in receive mode and similar dimensions.

Due to these issues, academia has produced quite a number of acoustic modems. However, they also have limitations, so we finally decided to build our own modem for swarms of  $\mu$ AUVs. In comparison with our modem, the WHOI micro modem 2 [27, 28] has a comparable consumption in receive mode, a higher consumption during transmission, but offers a hibernate mode below 1 mW. However, the micro modem is relatively large (1.6 times the length of our modem) for  $\mu$ AUVs and has a price comparable to the Evologics devices. Benson et al. use a home-made transducer for their UWmodem in [10]. While they achieve a very low unit cost of \$50, our modem is based on an off-the-shelf transducer involving no extra handcraft. Moreover, swapping the transducer of our modem to a custom one is relatively easy, as we will outline in Section 3. Their modem uses an FPGA as processing unit, whereas we rely on a microcontroller. We deliberately chose this option to increase implementation comfort (cf. Section 3.1). Our modem also consumes less power at comparable communication performance. Nowsheen et al. present a software-defined acoustic modem also based on an FPGA in Reference [50] with the demodulator only being available in Matlab. Demirors presented a similar but improved concept in Reference [18]. Both solutions provide high data rates of several kbit/s, but they are not available as stand-alone devices and are bulky and power-hungry due to the use of general-purpose FPGA development kits. The ITACA modem [70] is very promising for static and mobile underwater, low-power networks. However, the used piezo-electronic transducer is highly directional and the communication performance w.r.t. bit or packet error rates is unclear.

During past years, the research community presented a plethora of cheap, low-power, and small underwater modems. Indriyanto et al. built an acoustic underwater modem with an ultrasonic car distance sensor and an Atmega328 [35]. While being a notable approach, the high bit error rate

(BER) of 35% at 1.7 m distance currently excludes it from practical application. Zia et al. use a self-made transducer with a commercial piezo-ceramic cylinder, a Raspberry PI, and an Atmega328, plus a tone decoder for demodulation [90]. In both cases, system design provides cheap and low-power modems. However, published tests were done at extremely small distances (up to 2 m) only, too short for realistic  $\mu$ AUV use cases. Moros et al. evaluate new MAC protocols with the Nanomodem [49]. The latter uses binary chirp keying, has a diameter of 42 mm, 60 mm length, and can communicate at a 2 km range. Another small underwater modem (at 80 mm diameter and 100 mm length), which is used for  $\mu$ AUV communication, was presented by Tao et al. in References [37, 71]. Their modem has a 200 m communication range. Compared to our modem, the Nanomodem and the modem from Tao et al. have a low data rate of 40 bit/s and 55 bit/s, respectively.

At the conclusion of our research and study of existing devices, we realized the need to devise and develop an open modem platform with off-the-shelf components, published under a permissive open-source license.

### 2.3 Acoustic Underwater Communication and Signal Propagation

For acoustic underwater communication, an electric signal is converted to sound and vice versa; typically with a hydrophone or piezo-electric transducer. The transmit voltage response (TVR) is expressed as the resulting sound pressure level (SPL)  $L_{tx}$  vs. a reference pressure of 1  $\mu$ Pa per volt RMS input at a specified reference distance  $d_0$ . The free-field voltage sensitivity (FFVS) describes the output RMS voltage vs. a reference voltage of 1 V at 1  $\mu$ Pa of input pressure. Figure 9(a) and Figure 9(b) show TVR and FFVS of three AS-1 hydrophones [7]. By combination of both figures, it is possible to derive the output voltage at a receiver at distance  $d_0$  from the sender that sends at a given frequency  $f$  with input voltage  $V_{tx}$ . As an example, for an input signal with frequency  $f = 50$  kHz and voltage  $V_{tx} = 1$  V, the output signal at the receiver is approximately  $V_{rx} = 0.1$  mV. Depending on the type and characteristic of a hydrophone, FFVS and TVR are subject to directionality. However, in many use cases, proper installation and careful positioning of the hydrophones allows to neglect directionality to a wide degree.

Path loss depends on distance and the environment but is also affected by frequency-dependent absorption [14, 69]. A simplified attenuation model of an acoustic signal is

$$L(d, f) = \underbrace{20 \cdot n \cdot \log_{10}(d/d_0)}_{\text{spread loss } L_d} + \underbrace{(d - d_0) \cdot \alpha(f)}_{\text{absorption loss } L_f} \text{ dB,} \quad (1)$$

where  $L(d, f)$  is the attenuation of a sinusoidal signal with frequency  $f$  received at distance  $d$  from the sender versus the reference value  $L_0(f)$  at distance  $d_0$ . The path loss exponent  $n$  depends on the examined physical quantity and the environment; e.g.,  $n = 1$  for acoustic pressure under a free-field assumption and spherical spreading. This is the typical situation in underwater scenarios. In shallow waters, however, this assumption only holds for the line-of-sight (LOS) portion of the signal, whereas surface and other reflections contribute to divergence. Yet, Equation (1) yields a tractable approximation of expected receive signal level and system design for given communication ranges.

Several empirical models for  $\alpha(f)$  are discussed in, e.g., Reference [88] and are based on salinity, depth (or hydrostatical pressure, respectively), and water temperature. A relatively simple yet precise model has been suggested by Schulkin and Marsh. For the envisioned scenario in sweet water—or even in the nearby Baltic Sea (below 10 % salinity)—and communication ranges up to 100 meters, absorption is at most 3 dB in the shown frequency range.

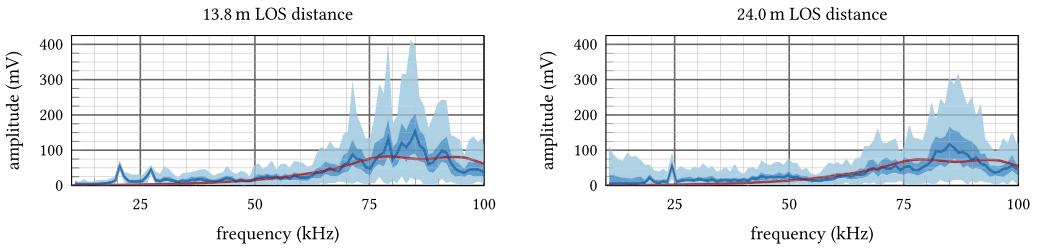


Fig. 3. Quartiles of received frequency shares of a 100 ms band-limited (10 kHz to 100 kHz) pseudo-random acoustic signal at two distances. Received signals were sampled at 1 MHz and broken into 98 segments of 1,024 samples each. Sender and receiver were submerged 0.8 m in LOS conditions in a marina in Hamburg. The pink line indicates the expected LOS response and considers path loss plus hydrophone and transmit/receive circuit transfer functions.

Finally, the SPL and the (RMS) voltage at the receiver can be expressed as

$$L_{\text{rx}} = L_{\text{tx}} - L(d, f) \quad \Rightarrow \quad V_{\text{rx}} = V_{\text{tx}} \cdot 10^{\frac{L_{\text{rx}}}{20}}. \quad (2)$$

Compositions of multiple sinusoids are obtained by superposition (the explanation is omitted due to page limitation). For the previous example, the output voltage at the receiver at  $d = 100$  m ( $10^\circ\text{C}$ , 10 % salinity) from the sender would be less than 1  $\mu\text{V}$  with ca. 41 dB path loss.

In contrast to radio communication, propagation delay of acoustic communication is high due to a low speed  $c$  of sound (in water) around 1400 m/s to 1550 m/s. As discussed in References [14, 88, 89], the actual value of  $c$  mainly depends on water temperature, salinity, and pressure. Several models for calculating  $c$  from these factors have been proposed, where the so-called UNESCO equation presented in Reference [89] is frequently used, e.g., in commercial underwater sensors [62].

Unfortunately, the presented models are simplifications only. In praxis, path loss is heavily influenced by other phenomena such as, e.g., directionality of the transducer (e.g., hydrophone), air bubbles, different water layers, turbulence, obstacles, and reflections (primarily at the water surface for shallow water communication). Figure 3 shows the influence of reflections (mainly at the surface) and their location dependency. Notches result from destructive inference, whereas spikes stem from constructive inference. In addition to these issues, the Doppler effect will lead to frequency shift when at least one of two communication partners is moving. The reader is directed to References [14, 69] for a more detailed discussion.

## 2.4 Modulation Techniques

As evident from Section 2.2, modulation in acoustic underwater communication is dominated by frequency-shift keying (FSK) and phase-shift keying (PSK). Differential PSK (DPSK) alleviates the tight synchronization requirements of PSK. However, PSK and DPSK require more resource- and computation-demanding demodulation techniques to assess the phase [41], particularly for carriers close to the Nyquist frequency. FSK, in contrast, can be demodulated efficiently in the transmission band with inexpensive, low-power hardware; even in case of mobility. A promising approach is to use chirp spread spectrum (CSS) for modulation—possibly in combination with FSK or PSK—to make communication more resilient. The advantages of CSS for underwater acoustics are discussed in References [40] and [66]; its main disadvantage is the computation complexity of demodulation. While modulation is an integral part of communication, we argue that our goal is not to present the best possible acoustic modem but a research platform that will enable and trigger practically backed research on, e.g., the best possible modulation scheme for underwater communication.

To elevate data rates, orthogonal frequency division multiplexing (OFDM) and multiple input multiple output (MIMO) for acoustics have been addressed in, e.g., References [13, 31, 51]. OFDM focuses around exploiting multiple carriers with low cross-talk (due to orthogonality) and is compatible to and realized in inexpensive low-power modems, including our modem. MIMO, however, demands for transducer (typically piezo) arrays—dramatically increasing cost and size—and additional computation capacity—demanding more capable hardware, power consumption, and size. MIMO is hence not considered in this work.

## 2.5 Underwater Localization

On top of communication, acoustic signals can be leveraged for distance measurements. As replacement of expensive standalone systems such as long baseline (LBL) and ultra-short baseline (USBL), acoustic modems can be employed. Additional benefits are the reduction of required infrastructure (by exploiting surfaced vehicles with a GNSS unit) and the avoidance of interference. The latter is typically achieved by exploiting regular data traffic (and the corresponding link-layer acknowledgments) to determine distances through two-way time-of-flight ranging (TWR).

As described in Reference [23], localization can be achieved through multi-lateration of several ranging measurements between a mobile node and anchors with known position. In Reference [9], Behrje et al. propose a controller that allows one or more submerged  $\mu$ AUVs to follow two surface vehicles based on TWR. Their approach is round-based and requires each submerged  $\mu$ AUV to determine the distance to the two surface vehicles once per round. The authors of Reference [21] use TWR in combination with stochastic gradient descent or a particle filter to find a target with a  $\mu$ AUV.

Another technique is to add angle information via hydrophone arrays, adding the drawback of more complex and costly hardware. This technique is usually applied in (sound) source localization [84]. A different approach is presented in Reference [20], where time of arrival between the spiral wavefronts of beacon signals is used to determine positions. This allows passive localization yet requires precise time-keeping by the receivers. For this purpose, the authors suggest to use chip-scale atomic clocks, which have almost 10 times the cost of one *ahoi* modem, so we do not consider this concept further.

## 3 SYSTEM OVERVIEW

The hardware/software co-design of an acoustic modem has an inherently large design space. In the following, we explain our fundamental considerations and present hardware implementation, software architecture, and communication stack of the *ahoi* modem.

### 3.1 Considerations and Fundamental Design Choices

Underwater sensor networks share with their above-surface counterparts that deployment strategies [8] are required, advocating the need for testbeds of several ten to hundred nodes [12] to overcome the gap between simulation and the real world.

Our main objective, hence, was to design a low-power, low-cost, miniature, open yet reliable device to enable practical research in underwater communication and localization protocols in shallow-water scenarios with communication ranges of ca. 100 m. Such a device is not available until today (cf. Section 2.2). We targeted employment in  $\mu$ AUVs such as MONSUN [48] and HippoCampus [30] and stationary, miniature underwater sensors. We strived for high customization, configuration, and modification potential, be it hardware or software, with the goal to provide an open, extendable, and inexpensive platform targeted but not limited to research. We tried to take all design decisions from the perspective of a researcher to make the device as flexible and versatile as possible. Finally, there is an extremely high degree of mutual impact of design choices, which

renders the design of a versatile research device difficult; e.g., the choice of the frequency band impacts sampling frequency, which impacts IC selection (price, power consumption, etc.).

We hence decided to implement most functionality in software and opted for a microcontroller ( $\mu$ C) as processing core, being a decent trade-off between versatility and low power consumption. We also assume that most users will have experience with  $\mu$ C programming, so the potential user base is extended over a system using a DSP or FPGA. All coding- and modulation-related functionality is realized in software, leaving a light burden of analog filtering and signal amplification as hardware realizations. Two positive side effects of this solution are the high degree of flexibility (most functionality can be changed in software) and the low effort and cost for assembly. Moreover, due to the modularity of our hardware, it is possible to add DSP or FPGA functionality later; plus, replacing the low-cost, low-power  $\mu$ C with a more powerful one is an option. It is even possible to omit the  $\mu$ C and connect the analog part of the modem to an oscilloscope or function generator; e.g., to carry out studies of the acoustic channel.

We opted for transmitter circuits without transformers to achieve a compact design and to keep transmit power consumption low at the cost of a higher receive-side amplification. With typical hydrophones and for the targeted communication range, amplification needs to be around 40 dB to 100 dB according to Equation (2). Due to limited range of affordable ADC ICs, an adjustable-gain amplifier in combination with automatic gain control (AGC) is required.

The communication frequency band needs to be matched to the hydrophone characteristics (e.g., maximum SPL, flat FFVS) and the required modulation scheme. The *ahoi* modem is compatible with various hydrophones and frequency bands. Adjustments are possible by exchanging a few capacitors (on the receiver module) and adapting look-up tables. The presented hardware and software are tailored for the AS-1 hydrophone (cf. Section 3.3). The default frequency band of 50 kHz to 75 kHz (with potential extension to 87.5 kHz) is a compromise of meeting contradicting demands, such as satisfying the Nyquist criterion and real-time signal processing at the same time.

Early experiments revealed that a combination of different techniques is required to counter the challenges of the shallow-water acoustic channel, as discussed in Section 2.3. In shallow water, multi-path propagation is a severe problem due to heavy reflections at the surface. Particularly at short distances of a few meters, (surface) echos lead to inter-symbol and even intra-symbol interference, so countermeasures are needed. Remedies are discussed and evaluated subsequently.  $\mu$ AUV movement leads to a frequency shift due to the Doppler effect; e.g., a 75 kHz signal will be shifted by 200 Hz if two  $\mu$ AUVs move at 2 m/s in the opposite direction. In contrast, at ranges up to 100 m and frequencies below 100 kHz, absorption is negligible with less than 1 dB in fresh water and at most 4 dB in sea water.

### 3.2 Acoustic Communication Stack

To enable basic communication and localization for  $\mu$ AUVs as part of our ongoing research [33, 64, 66], we implemented a reference communication stack, ranging from modulation to encoding. We discuss the stack subsequently and provide a detailed evaluation in Section 5 and Section 6. Due to the complexity of building a fully operational communication stack, we constrained ourselves to a choice of robust techniques, leaving optimization (e.g., other modulation techniques) to follow-up work; e.g., we are currently investigating chirp spread spectrum [66] and performed a comparison of modulation techniques in Reference [67]. Following this path allows research from modulation over medium access to localization but also employment in applications.

We implemented orthogonal binary frequency shift keying (BFSK). It is well suited for communication of moving devices and can be realized without tight synchronization between receiver and sender, keeping hard- and software less complex. Frequency spacing is 781.25 Hz with a symbol duration  $T_s$  of a multiple of 1.28 ms to achieve orthogonality. The latter is particularly useful to

mitigate the influence of the typical echos on the channel. For normal operation, we use a modulation index of 2 ( $T_s = 2.56$  ms) to improve separation of adjacent frequencies at the cost of a lower data rate. The frequency spacing also accounts for the Doppler effect at typical  $\mu$ AUV speeds of up to 1 m/s to 2 m/s for fast-moving robots. In the chosen setup, frequency deviation is below 250 Hz at 2 m/s and a 90 kHz sinusoid.

Signal composition and detection are done in software based on look-up tables. The receiver uses non-coherent detection. The sampling rate for sending and receiving is 200 kHz. To elevate the data rate,  $C$  bits are transmitted (concurrently) per symbol, which hence consists of  $C$  superimposed sinusoidal waveforms. This approach is more bandwidth-efficient than M-FSK at the cost of smaller amplitudes. The default is  $C = 4$  with an option to use  $C = 6$ . Larger values are possible, but will reduce signal-to-noise ratio per frequency and may break signal decoding in real time. Amplitudes of these symbol components are scaled by the sender to achieve equalized levels (amplitudes) at the receiver. We hence compensate the non-flat transfer functions of filter and hydrophone.

We employ frequency hopping (FHSS) to address inter-symbol interference. To counter frequency- and time-dependent attenuation due to multi-path propagation and reverberation, we added a redundancy mechanism that repeats each symbol ( $C$  bits)  $S$  times. From the  $S$  repetitions of each bit, the receiver makes its final decision based on the largest difference between detected amplitudes. This repetition in combination with FHSS distributes (spreads) bits (and symbols) across the channel without adding bandwidth by exploiting FHSS but by decreasing the data rate (by factor  $S$ ). This approach effectively advances packet reception rate (see Section 5.4) and comes at very low computation complexity, while it can be changed seamlessly at runtime.

Per-packet synchronization is achieved through a preamble of  $P_s$  symbols followed by a start frame delimiter (SFD). The receiver determines symbol windows after reception of  $P_r < P_s$  preamble symbols. Rather than simply alternating between two sine waveforms of different frequencies (space and mark), we address time- and frequency-dependent attenuation as follows: Each preamble symbol consists of two sinusoidal waveforms (typically a higher and a lower frequency from the used band) that are only reused every fourth symbol. These frequencies are not used for data transmission, and they are evenly distributed over the used frequency band. Default values are  $P_s = 16$ ,  $P_r = 11$ , and a four-symbol SFD, so the receiver has a second chance for synchronization.

In the default setup ( $S = 3$ ,  $C = 4$ ,  $T_s = 2.56$  ms), net data rate is 260 bit/s (520 bit/s gross) at a bandwidth of 25 kHz. The maximum data rate is currently 2.35 kbit/s net for  $S = 1$ ,  $C = 6$ ,  $T_s = 1.28$  ms, and a bandwidth of 37.5 kHz.

Due to the large, distance-dependent range of ca. 40 dB in expected signal amplitude (cf. Section 2.3), we implemented a lightweight automatic gain control (AGC) algorithm in software. It calculates the moving average of the rectified receive signal—we found that a window size of 320  $\mu$ s is a good trade-off between responsiveness and stability. If the average exceeds or undershoots a pre-defined target value plus or minus a threshold, then gain is decreased or increased (by one level), respectively. If AGC is disabled, then receive gain is fixed and can be adjusted manually.

Communication is packet-based. A packet consists of a six-byte header and the payload. The header includes addressing information, a sequence number, the payload length, and a flag field. The latter contains a flag to request an automatic acknowledgment and a flag to request a distance measurement based on two-way time-of-flight ranging. For details, we refer the reader to Reference [56] due to space limitations. We use extended Hamming codes, interleaving, and two checksums (CRC-8 for the packet header and CRC-16 for the payload).

For better understanding and illustration, Figure 4 shows an ideal packet (no path loss, absorption, etc.) from a receiver's perspective for the default setup of the communication stack. The spectrogram has been created with windows of length  $T_s$  and a 25% overlap. Artifacts stem from crisp changes of frequencies. In contrast, Figure 5 shows the spectrogram of a packet received in a small

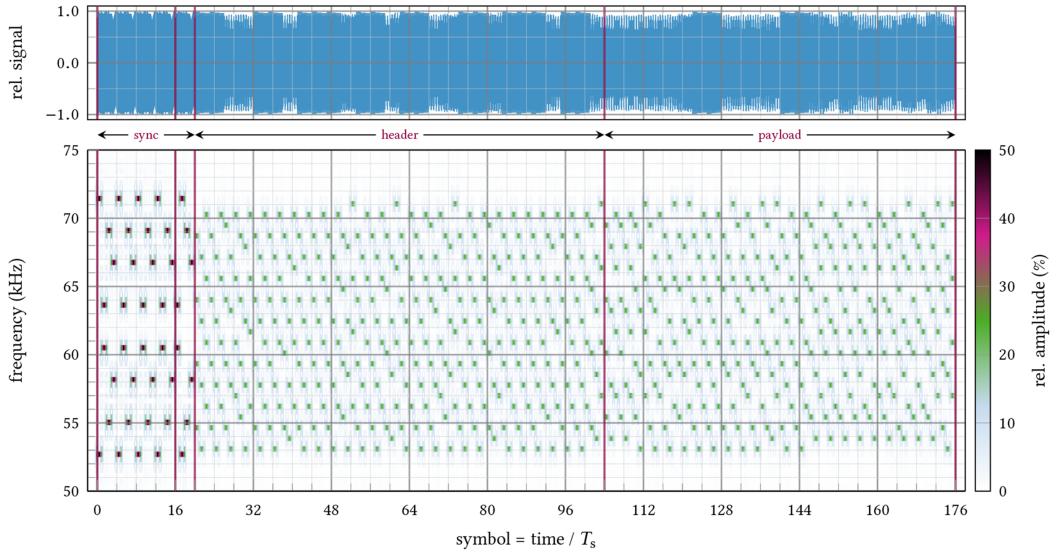


Fig. 4. Spectrogram of an ideal, received data packet with 4 B payload. The figure exhibits the different parts—preamble and SFD (labeled sync), header, payload—of the packet and illustrates the applied frequency hopping and parallel bit transmission. Preamble and SFD frequencies have double amplitude compared to header and payload due to different number of parallel sinusoids (two and four). Minor x-ticks separate groups of four symbols.

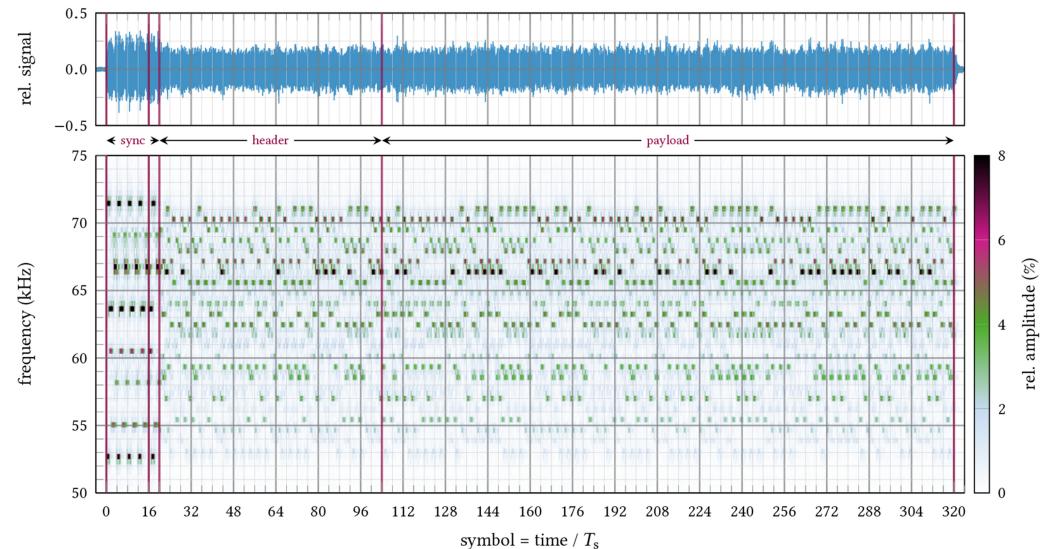


Fig. 5. Spectrogram of a received data packet (ca. 4 m  $\times$  2 m  $\times$  1.5 m). Display is analog to Figure 4, but the packet has a longer payload (16 B). Uneven distribution of amplitudes indicates the relevance of reflections, and why spreading is an effective countermeasure (cf. Section 5.4).

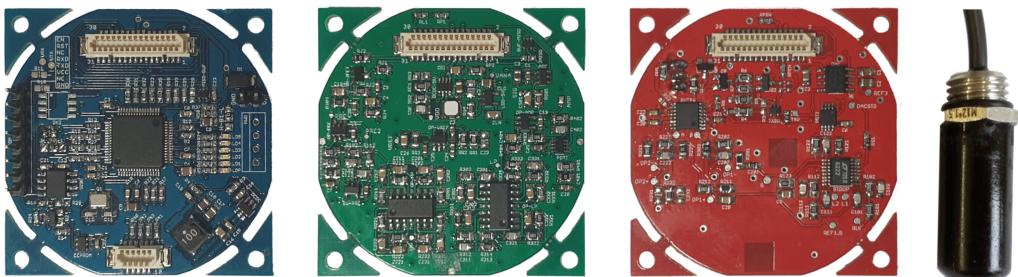


Fig. 6. *ahoi* modem (left to right): mainboard (blue), receive filter and amplifier (green), high-power transmit amplifier (red), hydrophone. Note that the power amplifier has half of its components on the back side. The hydrophone is a custom variant of an AS-1 with a threaded connector and is drawn to accurate scale.

tank environment, clearly showing interference (mainly due to reflections) and stressing the need for countermeasures as described previously. In particular, interference leads to selective cancellation of certain frequencies and, therefore, high bit error rates. Combining FHSS with bit repetition decreases bit errors and increases packet reception rates considerably (cf. Section 5.4). In contrast to a completely static tank environment, cancellation is highly dynamic in outdoor environments.

### 3.3 Hardware Implementation

Design of the hardware was an essential trade-off between (i) small size, (ii) modular design, and (iii) a reasonable price for low-budget research applications and commercial products. Ultimately, we adopted a rectangular shape with a length of 50 mm that can be reduced into a circular shape. Both shapes allow a direct integration in Hippocampus, which is one of the smallest general-purpose  $\mu$ AUVs. Modularity is achieved through a stacked approach, in which the three functional units (mainboard, receiver, transmitter) have a dedicated board or layer, respectively. Connection and communication between the boards is realized by a 31-pin Hirose DFS connector. Stack height of the assembled modem is a mere 25 mm. Figure 6 shows all parts of one *ahoi* modem. Input voltages range from 6 V to 16 V. Implementation details are provided subsequently.

The low-cost and small-size aspects of the system stem from several design choices:

- We used off-the-shelf components with most adequate specification and low prices.
- The transmit circuitry is kept at low complexity and is designed for and compatible with a variety of cheap transducers, which offer sufficient output power for moderate distances.
- We assume that the modem will be integrated into the  $\mu$ AUV or sensor node, hence eliminating any extra cost for expensive (titanium) housings; however, in many scenarios, a cheap watertight enclosure will suffice.
- We used a  $\mu$ C rather than FPGA or DSP; this reduces size, footprint, and the need for expert knowledge at the cost of lower processing power. The entire hardware abstraction layer of the used  $\mu$ C is available under an open-source license and, thanks to our software architecture (cf. Section 3.4), all algorithms and drivers can be reused for different  $\mu$ Cs, if desired. In contrast, FPGAs require vendor-specific tools that result in high long-term costs for licenses.
- All boards are two-layer designs with low component count and no extra-small packages, so they can be hand-soldered quickly and infer low production costs.

Finally, the modem has been developed as part of publicly funded research projects with a focus on scientific rather than commercial output. Therefore, we decided to make it open-source, so circuit layouts, firmware, and documentation will be freely available.

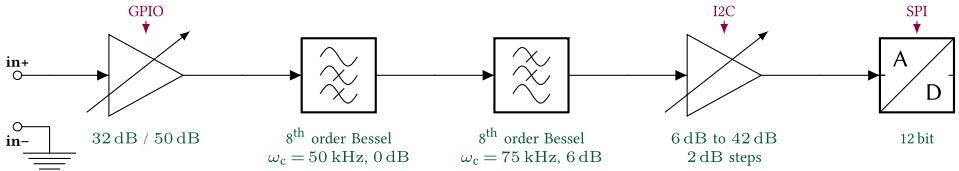


Fig. 7. Architecture of the receiver in default (frequency) configuration.

*Mainboard.* The mainboard holds the  $\mu$ C, main digital power supply, isolation ICs for serial communication with the host, and status LEDs. We chose a Cortex M4  $\mu$ C with a maximum clock speed of 180 MHz [68] to provide sufficient computing resources at an affordable price combined with low power consumption. To simplify debugging and evaluation, 11 GPIO pins are accessible through the connector. An efficient TPS 54202 [83] DC/DC voltage regulator provides the digital supply of 3.3 V for the  $\mu$ C and digital peripherals on other boards. Serial peripheral interface (SPI) and inter-integrated communication (I2C) are also available on the connector. Three dedicated GPIO pins are reserved and used to enable the receive and send circuits, respectively, and to switch between send and receive modes, so a single hydrophone can serve as sender and receiver. The mainboard also features a 2 kbit EEPROM for persistent storage of configuration data.

*Receive Filter and Amplifier.* The receiver board consists of the analog amplifiers and filters, a dedicated analog power supply, and an ADC. Our design uses a single-ended 5 V power supply—consisting of a TPS 62120 DC/DC converter [78] and a TPS 73150 LDO [73]—with a split rail (virtual ground at 2.5 V) provided by an AD8031 [3] op-amp. The board supports full shut-down.

Figure 7 shows the fundamental architecture. It consists of a JFET hydrophone driver and a pre-amplifier based on a low-power, high-bandwidth ADA4807-1 [5] op-amp with two electronically selectable gains via a switch-controlled gain path. A 16th order band pass filter reduces ambient and circuit noise due to the required high gain. High and low pass are realized in individual blocks to enable a modular design (e.g., independent change of corner frequencies via a single capacitor value) and the use of two different quadruple op-amps [76, 77] to cut costs and consumption. The final gain stage supports 18 gain levels in steps of 2 dB via an I<sup>2</sup>C-controlled potentiometer [2] in the feedback path of an OPA365 [81]. The latter combines a low price, noise, and offset voltage with a high bandwidth. However, the most striking feature of the IC is its extremely low bias current of only 10 pA, which helps to maintain a stable and small output offset error regardless of any input impedance mismatch due to the potentiometer.

Finally, a 12-bit ADC121S101 [80] is used. We picked this device because it combines a low price with high sampling rate (up to 1 MHz) and an easy electrical implementation: It has an internal reference and is compatible with 3.3 V logic from a single 5 V supply. In combination with our transmitter and the hydrophone, the amplification range is designed to allow communication from 1 m up to at least 200 m.

*Transmit Power Amplifier.* We designed two transmitters, a low-power device for shorter ranges up to ca. 100 m and a high-power device for longer distances. In the design of both amplifiers, we achieved a small size by avoiding the use of large transformers.

The low-power transmitter uses as output stage an OPA172 [82] op-amp. Power is supplied by an LTC 3265 [44] dual charge pump with two integrated LDOs, which provide  $\pm 18$  V at 50 mA at min. 11 V input. The supply is sufficient to drive all circuitry and the hydrophone at a maximum output amplitude of 15 V and at most 6 nF capacitive loading. The digital signal from the  $\mu$ C is converted by an AD5621 12-bit [4] DAC supplied by the 3.3 V from the mainboard. A 4th order low pass serves as reconstruction filter. A digitally controllable high-voltage analog switch IC [6] in conjunction with a low on-resistance JFET connects the hydrophone to either the send or receive

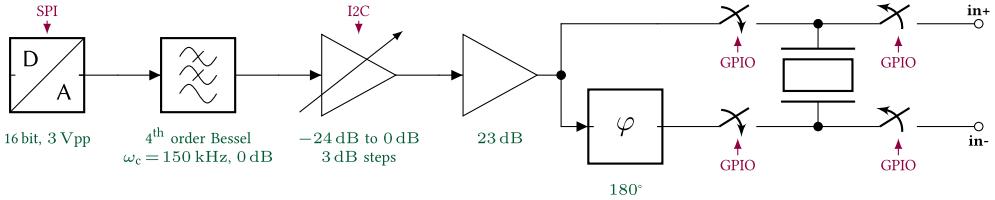


Fig. 8. Architecture of bridge transmitter.

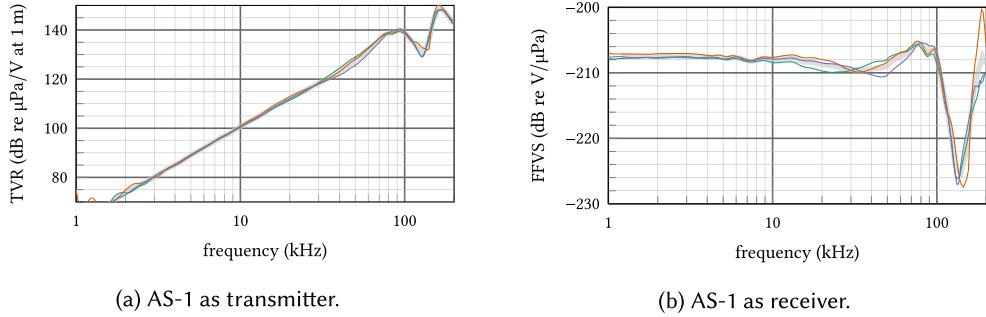


Fig. 9. Transmit Voltage Response (TVR) and Free-Field Voltage Sensitivity (FFVS) of three Aquarian Audio AS-1 hydrophones [7]. Voltages are RMS values.

circuitry. The switching circuit resides on the power amplifier, because it requires full (output) supply during transmission. The amplifier circuit is disabled when it is not used and only consumes a few microwatts.

The high-power transmitter uses an output driver consisting of an OPA551/2 [79] pair of opamps in bridge topology, yielding a maximum output amplitude of ca. 40 V at 200 mA peak current. This gives an elevated communication range of factor 2.7 or 8 dB, respectively, at a higher baseline consumption (cf. Section 4.2). The power supply consists of a boost converter (LMR61040 [74]) and a buck converter in inverting topology (LMR16006 [75]). Output voltages are  $\pm 22$  V with a few hundred mA output current. The amplifier features an electronically adjustable output level in nine steps of 3 dB and a shut-down mode. Like its sibling, it has a switch to connect the hydrophone to either send or receive circuitry with a different IC [46] due to the bridge topology in Figure 8.

*Transducer.* We decided on an Aquarian Audio miniature AS-1 hydrophone [7] acting as both sender and receiver as default setup. It has a very small size ( $\varnothing 12$  mm  $\times$  40 mm), a still affordable price of \$400, and was readily available in small numbers. It offers a high bandwidth of almost 100 kHz with a flat FFVS and a relatively high, quasi-linear TVR (see Figure 9). The AS-1 has good omni-directionality, which is critical for swarms of AUVs. It has a capacitance of ca. 6 nF and tolerates input voltages up to 150 V (peak-to-peak). Here, we want to point out that our modem can be used with many hydrophones available on the market, depending on the purpose and budget. The communication frequency band can be adjusted through a few capacitors (on the receiver board) and by modifying the look-up tables, and the transmit-receive-response can be configured via the coefficients for gain compensation (cf. Section 3.2).

### 3.4 Software Architecture and Communication Interface

The software architecture enables the development and replacement of software and hardware components through a layered approach (see Figure 10). The main goal is to allow researchers to:

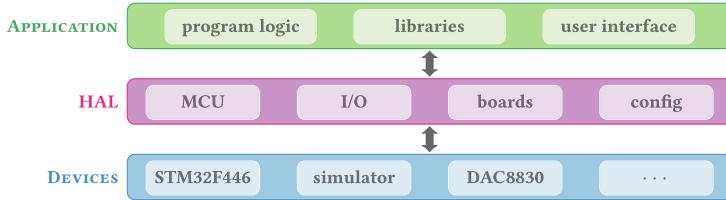


Fig. 10. Modular software architecture to split core functionality and device drivers, connected by HAL layer.

- replace all software components (e.g., encoding or modulation) transparently to carry out research on the different aspects of underwater communication,
- revise, adapt, or add boards with little software development effort, and
- use all algorithms on other hardware or even for simulation.

All hardware-independent software—including the program logic, libraries, and the user interface—resides in the application layer. The hardware abstraction layer (HAL) defines interfaces for hardware-dependent software for, e.g.,  $\mu$ C, IO, and individual boards (cf. Section 3.3). Device drivers,  $\mu$ C specifics, and a simulation environment are implemented as part of the device layer. The selection of the appropriate modules for a specific target platform and board revisions is handled through make targets at compile time. This architecture allows the use of the same application code for different board revisions,  $\mu$ Cs, and even a simulation environment, while it enables reuse of device drivers for different boards.

Communication between host ( $\mu$ AUV) and modem is packet-based (UART) and transparent—i.e., packet format for acoustic and serial communication is identical. Packets sent from host to modem are immediately put on the acoustic channel, and packets received by the modem are immediately forwarded to the host. Data-link layer services such as back-offing, retries, and so on, are done by the host or can be implemented on top of the current communication stack. We defined a set of special packets (types) that are processed by the modem (and not transmitted on the acoustic channel). They act as commands to read out values, such as the current noise level or packet statistics, and to change parameters such as the gain level,  $S$ , and  $C$ . They hence allow for easy control of the modem and parameter change for conducting experiments.

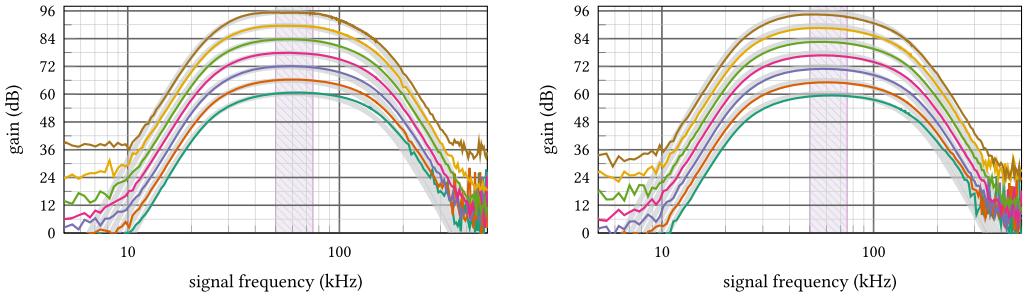
## 4 HARDWARE EVALUATION

We assessed and validated the hardware with focus on consumption and quality of receive filter chain and power amplifier. For this purpose, we built and analyzed six mainboards (MB), eight receivers (RX), seven default (TX), and seven bridge (TXB) transmitters.

### 4.1 Receiver Analysis

*Test Setup.* We validated gain and filter characteristics of our receiver through laboratory tests. First, we determined the transfer function from receiver input to output (ADC input). Second, we connected a hydrophone to the receiver input and measured the noise level at the output. We conducted all measurements [...] for all gain setups of the second stage (and the first stage fixed at the higher gain). We repeated both measurements for one receiver connected to various transmitters to assess the influence of signal path switching.

With the chosen setup, the expected gain ranges from 60 dB to 96 dB and an anticipated communication range of more than 100 m and no signal clipping at distances around 2 m (for the lowest gain level). All measurements were done with a TiePie Handyscope HS5, which we programmed to feed sinusoidal signals to a voltage divider with signal attenuation of 100 dB to obtain input



(a) Direct input to receiver (RX-8). (b) Input to receiver (RX-8) via transmitter (TX-1).

Fig. 11. Transfer functions of a receiver fed directly and through a transmitter for every third gain level (6 dB steps) starting at the minimum setting. Same colors indicate same gain setting. Intermediate gain levels are omitted for clarity of display. The frequency band typically used for communication is highlighted in pink; ideal transfer functions are displayed in bold gray behind measured curves.

voltages expected in real-world scenarios (cf. Section 2.3). The output of the voltage divider was connected to the receiver/transmitter input via a 5.6 nF<sup>2</sup> decoupling capacitor. A Python script ran all tests automatically by setting the receiver gain and adjusting the function generator output to obtain a target voltage at the ADC input of roughly 2 V to 3 V peak-to-peak amplitude in the passband. The script subsequently generated sinusoidal signals with frequencies from 0 kHz to 500 kHz with step size of 390.625 Hz (half the frequency spacing of the FSK; cf. Section 3.2) for frequencies below 100 kHz and 6.25 kHz for frequencies above 100 kHz. All test signals had a duration of 2.56 ms with a short 0.2 ms period of silence (0 V DC output) before and after the signal. Function generator and scopes ran a sampling rate of 5 MHz. We determined per-frequency gain by cross-correlating input and output signals scaled by the input attenuation. Average gain was obtained in the passband. We calculated noise figures in terms of AC RMS output, with 14 bit-resolution and a 1 V range.

*Transfer Function.* Figure 11 shows the transfer functions of one receiver. Results are representative. The figures clearly indicate the proper shape of the transfer function w.r.t. ideal behavior. There is some deviation in the upper cut-off region (the actual low pass has a slightly higher cut-off than the ideal one), which is due to capacitor tolerances (we used 5% parts and confirmed a deviation from nominal values of a few percent). However, the noted deviation is well inside the anticipated deviation shown by the used filter design tool. The effect is smaller for the high-pass portion, where actual capacitances meet the nominal ones better (we also confirmed this by measurement). The gain-bandwidth product of the op-amp in the final amplifier stage reduces the bandwidth at high gains and hence limits the maximum achievable gain.

The figures also reveal that the true 3 dB cut-off frequencies are shifted due to the concatenation of a high and low pass filter. As a result, the passband is wider than generally desirable but also flatter. For frequencies below 10 kHz, the transfer function is effectively clipped due to circuit noise in combination with the available scope resolution.

The influence of the transmitter is marginal, as indicated by Figure 11(b). Overall amplification is slightly decreased, likely caused by parasitic resistance and capacitance. For high gains, a low-pass characteristic is visible, leading to a small but harmless negative slope in the passband.

*Gain and Noise.* From the transfer functions, we calculated average gain in the communication band, displayed in Figure 12(a). The figure exhibits low deviation across individual receivers and

<sup>2</sup>Approximately the output impedance (capacitance) of the hydrophone.

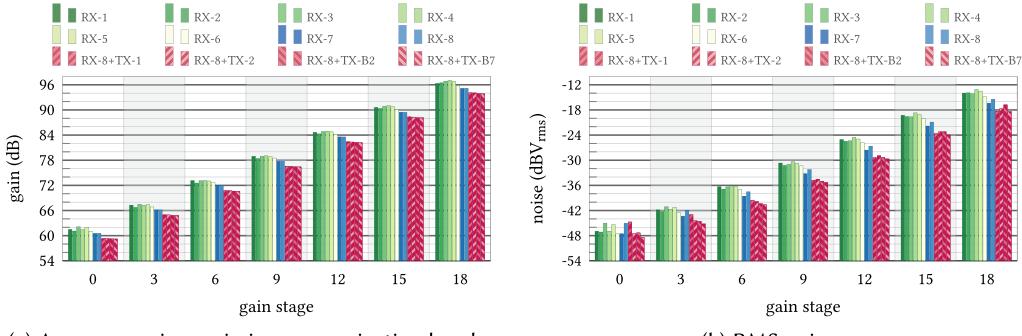


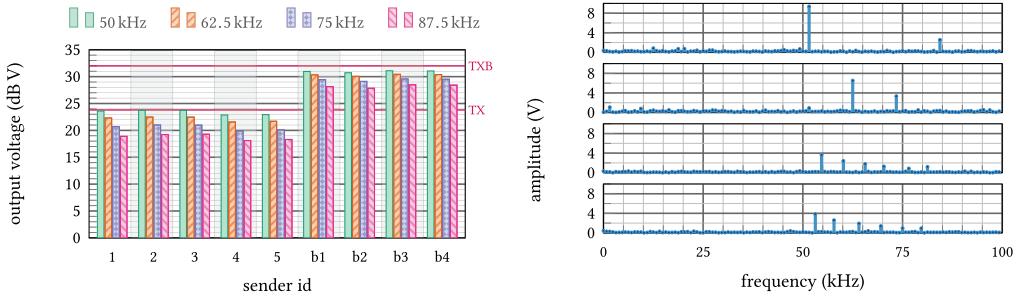
Fig. 12. Gain and noise study of all prototype receivers with a gain stepping of 6 dB (three steps, same setting as in Figure 11). Figures of double-sided receivers are solid turquoise; those for single-sided receivers are blue. Logarithmic y-axes for better comparability. For reference, a rail-to-rail sinusoid at the output would have 5 dBV<sub>RMS</sub>. The noise floor of the measurement device is -61 dBV<sub>RMS</sub>.

a consistent amplification characteristic with small offset below 2 dB. It is notable that gain steps are close to the desired 6 dB (of step size three) with small decay only—in the highest gain setting, they fall roughly 1 dB short. With transmitter connected, gain is slightly reduced by 1 dB to 2 dB but very consistent across gain levels and different transmitter boards.

Figure 12(b) shows noise figures in logarithmic scale. Noise is consistent across different receivers and gain levels and it follows the 6 dB steps with a tendency of an increased gain-to-noise ratio for large gain settings, because at higher gains, the final amplifier exhibits a lowpass characteristic—its cut-off frequency decreases with increasing gain. With transmitter connected, noise is reduced to a similar extent as gain, meaning that signal-to-noise ratio is unaffected. A closer look at the noise spectrum (not shown) identifies the analog supply and the SPI chip select as major cause of noise. While the LDO effectively reduces overall supply noise, the relatively low switching frequency of the converter (slightly above 100 kHz) is visible in the spectrum. Spikes at multiples of 100 kHz stem from the SPI chip select line. While there is likely room for mitigation, noise is acceptable.

## 4.2 Transmitter Analysis

Quality of transmission is an essential factor for successful communication, so we analyzed the output of transmitters with a connected AS-1 hydrophone via an Agilent MSOX3014T oscilloscope. Output voltage was measured directly at the output(s) of the transmitters, i.e., across the hydrophone. Sinusoidal test signals were created by the modem at maximum amplitude. Figure 13(a) portrays the results for three frequencies in the communication band. Output voltages among devices of same type are consistent with variation of at most 1 dB (for TX-4 and TX-5), which stem from component variation (we used a new batch of resistors in the amplifying part). At 50 kHz, all output voltages are within 1 dB of theoretical values. For the bridge amplifier, the gap is slightly larger mainly due to higher losses in the receive/transmit switch caused by the larger output voltage and current. Decaying output voltage with increasing frequency is expected due to the reconstruction (low-pass) filter by 1 dB to 2 dB. The bridge amplifier produces clean sinusoidal outputs with the expected attenuation. In case of the low-power transmitter, however, we observed -3 dB at 75 kHz and almost -5 dB at 87.5 kHz, respectively. The reason is that we operate the power supply close to its maximum current output and the DAC close to its physical limits w.r.t. settling time and slew rate. However, if the transmitted signal consists of multiple sinusoids—as for our



(a) Output voltage levels (amplitude) of five low-power transmitters and four high-power bridge transmitters (maximum gain) for sinusoidal inputs. Magenta lines indicate theoretical output level. (b) Output voltage amplitudes (with equalization) of two preamble symbols (top two rows, two sinusoids each) and data symbols (bottom two rows,  $C = 6$ ) of an actual transmission.

Fig. 13. Study of output levels and quality for low-power and high-power transmitters.

transmission scheme in Section 3.2—artifacts become insignificant and the low-power transmitter is usable without restriction. Figure 13(b) shows the spectrum of two symbols from a preamble and a data packet. Each preamble symbol contains two sinusoids and each data symbol contains six ( $C = 6$ ). The figure shows that, in praxis, there are no artifacts in terms of dominant frequency shares other than expected. Amplitude decay is due to equalization.

#### 4.3 Power Consumption

A major design aspect was to achieve low overall power consumption, so the *ahoi* modem can be powered by either a small, dedicated (rechargeable) battery or from the  $\mu$ AUV’s main supply without affecting mission time. To understand power consumption in more detail, we analyzed the components individually. Unless otherwise noted, we measured the supply voltage and the current consumption with two Voltcraft VC270 multimeters to calculate power consumption. All measurements include losses introduced by switching converters.

*Mainboard.* Power consumption of six analyzed mainboards ranges from 195 mW to 205 mW with a mean of 200 mW in listen mode at a sampling frequency of 200 kHz. Considering that all signal processing is done by the  $\mu$ C, power consumption falls well within scope of commercial modems (cf. Section 2.2). Consumption during reception and transmission is influenced by the number of parallel bits  $C$  but is similar to consumption during listen mode. Since listening is predominant and transmission power is dominated by consumption of the transmitter board, we did not inspect the exact consumption of the mainboard for these states in more detail.

*Receiver.* Power consumption of the receivers was measured in sleep and listen modes. A hydrophone was connected to the receiver, and gain was at level 3 (66 dB). Through additional measurements, we confirmed that the gain level influences consumption merely, because the use of the potentiometer does not change the load impedance (at the output) of the corresponding gain stage. Average power consumption is 99 mW, ranging from 98 mW to 101 mW among six boards. Sleep consumption is below 0.2 mW in all cases and hence negligible compared to consumption of a  $\mu$ AUV that consumes several watts even when not submerged. Consumption figures show that the receiver is extremely low-power at decent filter and gain performance. Only one-third of overall consumption in listen mode is due to the receiver.

*Transmitter.* For both transmitter types, consumption in sleep mode is below 1 mW and hence negligible when the modem is in listen mode. In idle mode, however, consumption of the

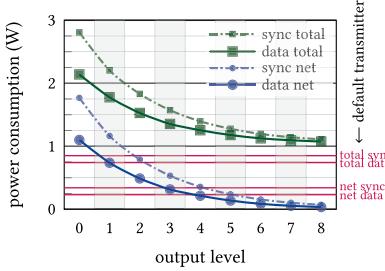


Fig. 14. Power consumption of a bridge transmitter vs. output levels during transmission of the preamble (two sinusoids) and the payload (four sinusoids). Corresponding values of a low-power transmitter are displayed on the second y-axis for comparison. Total values are consumption of a modem (including mainboard and receiver in sleep mode); net values are excluding mainboard and transmitter idle consumption.

low-power transmitter rises to 339 mW on average, whereas the bridge amplifier consumes 791 mW, ranging from 768 mW to 821 mW due to tolerances.

Next, we analyzed power consumption during transmission with an AS-1 hydrophone connected and submerged in water for one prototype of each transmitter type. We sent a short packet with activated equalization while we recorded the supply voltage and current consumption across a  $2.1\ \Omega$  shunt with an Agilent MSOX3014T oscilloscope. From these traces, we calculated average power consumption during transmission of the preamble (two sinusoids) and payload (four sinusoids). Figure 14 portrays average consumption of the modem and the transmitters' split. Total consumption increases from 1.1 W (lowest output level) to 2.1 W (highest output level) for the bridge amplifier during data transmission, whereas the low-power amplifier consumes 740 mW. For short-range applications (up to 100 m; see Section 5.7), the low-power transmitter is hence the better choice from a power-efficiency perspective. The increased range hence comes at a notable consumption penalty of up to 190% and 230% for data and preamble transmission, respectively. Only a small fraction of this power is due to the increased output of the hydrophone. For output level 3, at which the output of both transmitter types is almost equal, the penalty is still 83% and 85%, respectively.

*Summary.* Total consumption in idle and receive mode is 300 mW on average and a few milliwatts in sleep mode, so the *ahoi* modem is on par with commercial devices. During transmission, consumption rises to 2.1 W and 750 mW, depending on the transmit board used. For ranges of a few hundred meters, we believe the higher consumption is tolerable. With these figures and a typical mission duration of up to 5 h, it is possible to run our modem in listening/receiving mode with a LiPo-battery rated at 11.1 V with as low as 135 mA h capacity. With a battery of 1000 mA h at 11.1 V and a mixed-mode operation—e.g., one short transmission every 30 s, which is roughly equal to 5% transmit time—lifetime is approximately 28 h for the bridge transmitter and 34 h with the low-power transmitter. When using the main supply of a  $\mu$ AUV such as MONSUN with a 4.5 A h LiPo-battery at 11.1 V in a 5 h mission, the modem will only use 4% of its host's energy budget. With a dedicated battery for the modem, 175 mA h capacity would be sufficient. Figures for the low-power transmitter are 17% smaller.

#### 4.4 Processing Delays

Signal processing is done in real-time. All samples fed to the DAC are computed on the fly. During reception, each symbol is decoded before the end of the following symbol. Final packet processing requires 46  $\mu$ s (for the header) plus an additional 5  $\mu$ s per byte of payload; e.g., a packet of 96 B

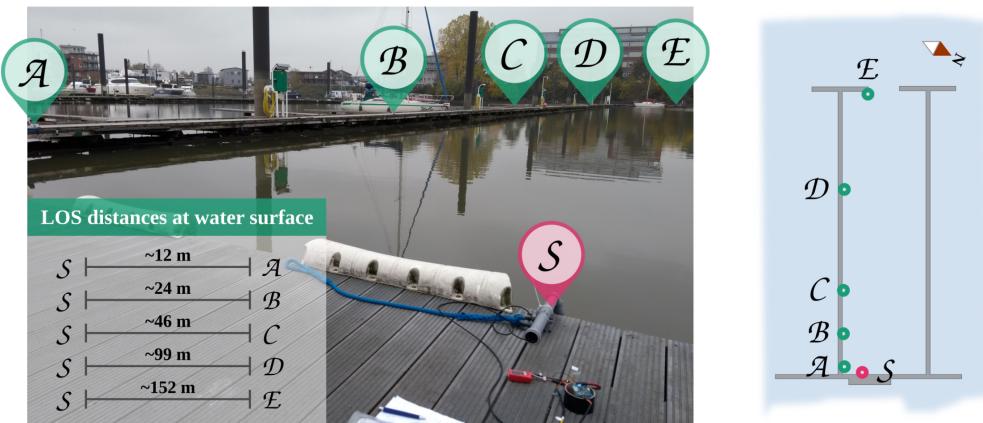


Fig. 15. Finkenwerder setup for communication tests with five receivers ( $\mathcal{A}$  to  $\mathcal{E}$ ) and one sender  $\mathcal{S}$ .

payload is decoded in  $520 \mu\text{s}$ , which is considerably below the used TX/RX turnaround time of  $12 \text{ ms}$  (to stabilize power supplies and safe switching of the hydrophone connection from input to output mode and vice versa). For packet reception, we use a small buffer of  $256$  samples to compensate for short lags; e.g., when decoding a symbol. Each modem records buffer overruns; in the experiments we conducted, we never saw this happen during transmission or reception.

## 5 REAL-WORLD COMMUNICATION EXPERIMENTS

We report on real-world experiences with our acoustic modem obtained in extensive communication experiments in a static scenario. Quality of communication in mobile cases is evaluated in Section 6. Our study is intended to show the general functionality of our modem, explain real-world phenomena, and match expectation to real-world observations. We also aim at showcasing research directions for underwater communication and algorithm development.

### 5.1 Experiment Setup

After days and hours of preliminary tests, we ran a full-day outdoor experiment in the marina of TuS Finkenwerder in southern Hamburg. The marina is located in a branch of the river Elbe, where depth ranges from ca.  $3.5 \text{ m}$  to  $7.5 \text{ m}$ , depending on the tide. We conducted our experiments on November 11, 2018, from 10 am to 6 pm with high water around 2 pm. Water temperature was  $9.9^\circ\text{C}$ , and we noted rare and little surface waves. Salinity was  $0.68 \text{ ‰}$ , resulting in a speed of sound of  $c = 1447.7 \text{ m/s}$ . Environmental data were recorded with a professional CTD-48 probe manufactured by Sea&Sun Technologies [62].

We deployed six modems in total, of which one acted as sender (position  $\mathcal{S}$ ) and the other five as receivers (positions  $\mathcal{A}$  to  $\mathcal{E}$ ). Positions and LOS distances are depicted in Figure 15. We arranged the modems with a tapeline and measured precise LOS distances with a Leica Disto A5 laser measurement device. The electronics of the modems were deployed on the jetties, and the hydrophones were submerged  $1.5 \text{ m}$  below the surface. All modems were connected to a laptop each via the serial line, and the receiver laptops were controlled via Wi-Fi from the sender laptop. Unless otherwise noted,

- the sender at pos.  $\mathcal{S}$  sent 200 packets with a payload of  $8 \text{ B}$  every  $750 \text{ ms}$  (allowing for a  $150 \text{ ms}$  channel cooldown phase between transmissions);

- we used the default frequency setup as elaborated in Section 3.2 (32 orthogonal, equi-distant frequencies in the band  $B$  from 50 kHz to 75 kHz with  $T_s = 2.56$  ms and  $S = 3$ ), and
- the modems were equipped with the bridge transmitter.

We used a preamble length of  $P_s = 16$  symbols, of which the receiver was required to successfully detect  $P_r = 11$  symbols before running the synchronization algorithm and waiting for the SFD. If enabled, then the automatic gain control (AGC) was set up with a target level of 25% (rectifying averaging) during reception with a threshold of 0.6 and 1.6 ( $\pm 4$  dB or two gain steps, respectively). These parameters were determined empirically in earlier experiments. In most conditions, this setup led to a relative signal amplitude of 30%–50% during reception, leaving enough headroom in case of short-term amplitude variation due to noise and reflections to prevent overdrive while providing sufficient resolution for successful decoding. In all experiments, we applied transmit equalization based on the frequency response of transmit and receive circuits plus hydrophone. We did not incorporate frequency-related absorption due to its low impact at short distances.

Please note that results are in logical argumentation order rather than timely order. Unfortunately, some results are not available due to depleted laptop batteries, because the tests ran for more than 8 h; too much for some of our laptops. We did not repeat those tests on a different day or do tests on different days to begin with, since it is already difficult to compare and match results from a single day due to changing environment and conditions, as we will show subsequently.

## 5.2 Metrics and Recorded Data

The main metric used in the following evaluation is packet reception rate (PRR), which indicates the percentile of received vs. sent packets. We use this metric to quantify overall communication quality and to compare different setups and configurations. We also recorded normalized loudness and received-signal-strength indicator (RSSI) for each packet. The former is defined as the square root of the (mean-free) signal power divided by the signal power of a (mean-free) perfect signal (full range, no noise, all frequency components having equal amplitude). Technically, this value is reported as a percentile by an unsigned 8-bit integer, so its range is limited to 255% (values exceeding this limit are clipped) and its resolution is 1%. Please note that in case of receiver saturation or overdrive, the (analog) signal is clipped, implying that values above 100% are lower than they would be without range limitation. RSSI is the mean relative amplitude of decoded signal (bits) and reported in percent. In case of spreading, only the strongest signal (bit) is used, which corresponds to the one that is used for demodulation decision. In case of a noise-free signal with uniform receive amplitudes, RSSI and loudness are equal. In case of noise and echos, loudness is typically higher than RSSI; in case of a heavily non-uniform receive spectrum, RSSI may exceed loudness. Please note that we display loudness and RSSI relative to 100% in decibels (i.e., 100%  $\hat{=} 0$  dB) for improved clarity of results.

In each experiment, all modems counted the number of synchronizations (preamble detections), SFDs (following a preamble) and successfully received packets. They kept track of the preamble detection process by counting the number of subsequently detected preamble symbols (until reception of a complete preamble or until no further preamble symbol was detected) to analyze the synchronization process and to judge the choice of preamble length. Here, we want to point out that after a failed synchronization or SFD detection, the subsequent signal(s) on the channel for packet header and payload may produce false positives w.r.t. preamble symbol detection due to frequency cross-talk. In general, the probability of seeing a complete preamble including SFD during a packet transmission is low. If a complete preamble (including SFD) would be detected falsely, then this condition would be handled during packet reception in terms of excessive bit errors or a bad checksum. This implies that fully decoded packets are (with very high probability) valid.

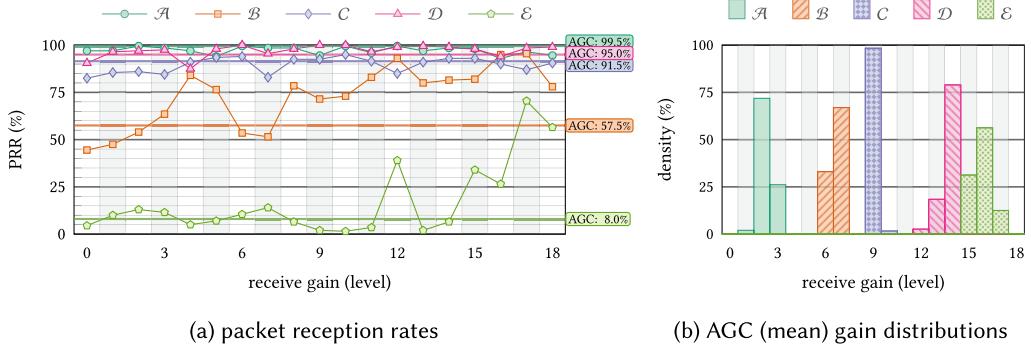


Fig. 16. Detailed study of PRR for all receiver gain levels and AGC accompanied by gain level distribution for AGC. Note that there is no overlap of bars for positions  $\mathcal{D}$  and  $\mathcal{E}$  in Figure 16(b).

### 5.3 Distance, Position, and Receive Gain

First, we studied the influence of distance, position, and receive gain level on PRR. Figure 16 portrays the results for all receiver positions in terms of PRR and average gain distribution for AGC.

*Packet Reception Rate.* For positions  $\mathcal{A}$ ,  $\mathcal{C}$ , and  $\mathcal{D}$ , PRR is above 85% in all but two (out of 57) cases, as can be seen in Figure 16(a). For position  $\mathcal{A}$ , 3,887 out of 4,000 (or 97.1%) packets across all gain settings have been received. Variation across gain settings is low, and AGC performs on par with fixed gains. Observations for positions  $\mathcal{C}$  and  $\mathcal{D}$  are similar with slightly lower PRR. As expected, PRR generally decreases with distance.

For position  $\mathcal{B}$ , however, PRR is much lower despite its second-shortest distance. There also is a notable variation and increase of PRR for higher receive gain. Results for position  $\mathcal{E}$  exhibit a similar behavior. PRR is around 10% in many cases with a spike for gain level 12 and improved PRR starting at gain level 15. Due to the lack of signal traces, clear identification of the reasons for these observations is impossible. However, the latter positions are different in that they are close to a boat, potentially causing more reflections and scattering of the signal. Another reason may be cancellation due to a poor alignment of LOS and surface-reflected signal paths. Apparently, PRR still increases with receive gain at these positions due to a higher signal level (and higher SNR in terms of ADC resolution). What is surprising at first glance is the poor performance of AGC, the steep increase of PRR for high gain levels (15 and above) at position  $\mathcal{E}$ , and the spikes and notches for some gain levels in both traces. Apparently, due to underwater currents, changing wave pattern, and other environmental aspects (likely including increasing water depth during the experiment), channel quality is massively changing at these positions. What is particularly hinting at this, is that the very first measurement was done using AGC, followed by fixed gains in increasing order. It is therefore traceable that AGC is on par with low gains because of timely proximity. In follow-up experiments, we repeated the measurement with AGC as a baseline, where results were departing massively (e.g., cf. Figure 18(a), Figure 18(b), and Figure 21(a)). We hence conclude that in certain positions, communication is extremely stable, whereas it is highly fluctuating in others; which is totally in line with other findings regarding shallow-water communication, e.g., References [15, 85].

Figure 16(b) portrays the distribution of average gain levels during packet reception with AGC. Variation among packets is very low with at most three gain levels at one distance. The recorded data also exhibit (not shown in the figure) that variation of the gain level during reception of a single packet never exceeds one gain level in each direction, indicating that both signal level (loudness) and AGC are stable. The average gain level from positions  $\mathcal{A}$  to  $\mathcal{D}$  increases by roughly

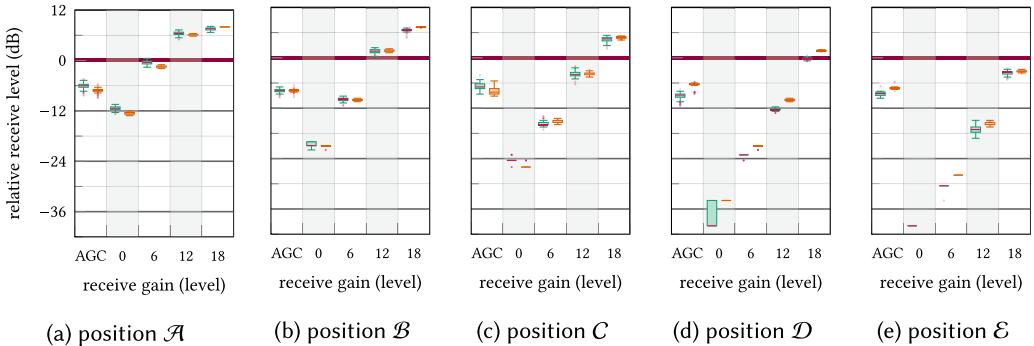


Fig. 17. Relative RSSI (left, green boxes) and signal loudness readings of successfully received packets for different distances and receive gains. Box plots indicate median, quartiles, and outliers. Values are relative to the maximum value at a perfect signal with full amplitude, where values above 100% (0 dB) (pink horizontal marking) are subject to signal clipping and are only shown for completeness. Major y-ticks are equivalent to expected difference between gain levels on the x-axis.

three steps each, which translates to 6 dB (or an amplification factor of two) and is in line with expectation at doubled distances.

We hence conclude that AGC fulfills its purpose and can effectively set the target signal level to react to environmental noise and different reflection scenarios: In noisy conditions, e.g., AGC will prevent signal clipping during reception. However, the results do not permit a conclusion on whether AGC has the potential to elevate PRR.

*Signal Receive Level and RSSI.* Along the lines of the evaluation of PRR vs. gain level, we took a closer look at loudness and RSSI values of received packets. Figure 17 exhibits the corresponding results via box plots with the following main findings: First, for values below 100% or 0 dB, respectively (i.e., non-saturated signals, no or low clipping), loudness and RSSI follow the expected behavior regarding distance (cf. Section 2.3) and gain levels (six gain levels correspond to 12 dB in anticipated loudness/RSSI). Second, loudness and RSSI values are typically on par. In some cases (e.g., at positions  $\mathcal{D}$  and  $\mathcal{E}$ ), loudness is notably higher than RSSI, indicating noise. In the concrete scenario, we expect the latter to stem from reflections. At position  $\mathcal{A}$ , RSSI is exceeding loudness, hinting at signal cancellation or an uneven distribution of signal or symbol amplitudes, respectively. However, there is no clear and deterministic relation evolving from loudness and RSSI imbalance on one hand vs. PRR on the other hand. Third, signal loudness at position  $\mathcal{E}$  in the largest gain setting is around 65% (ca. -3 dB), while other recordings indicate that reception with values as low as 1–2% is possible. This leads to the conclusion that the communication range of our modem may well be extended, and it will be a topic we will look into in the future. This also implies that long-range communication is achievable with low power consumption of few watts. Fourth, the AGC adjusts signal loudness to values between 25% and 60% (ca. -12 dB to -4 dB) in most cases. This falls in the desired range w.r.t. linking the different metrics for AGC (rectifying averaging) and loudness (RMS). Unfortunately, the interpretation of results is not straightforward due to these different metrics. Yet, the results support that AGC achieves reasonable signal amplification in different scenarios and at different distances.

#### 5.4 Symbol Repetition with Implicit Spreading

Due to intra- and inter-symbol interference in shallow waters—as outlined in Section 2.3—we apply the spread-spectrum technique sketched in Section 3.2. To assess the effectiveness of this method,

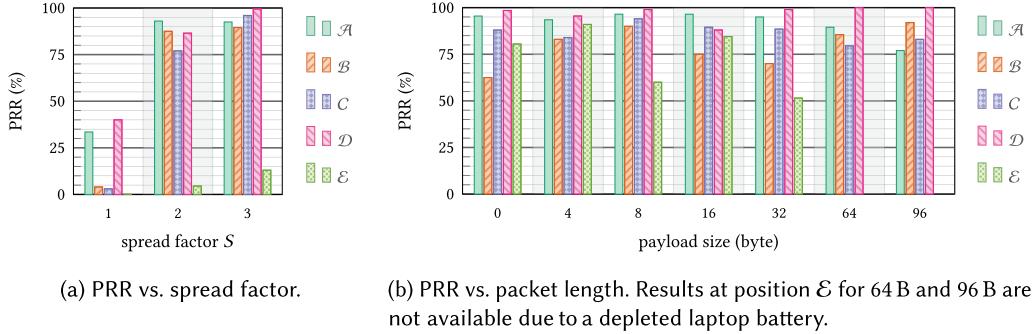


Fig. 18. Influence of spread factor and payload length on PRR at different distances with active AGC.

we studied the impact of  $S$  on PRR. Figure 18(a) portrays the results for  $S \in \{1, 2, 3\}$  for all positions using AGC and the default configuration. The figure reveals a benefit of more than 100% when increasing  $S$  from 1 (no spread-spectrum) to 2 (each bit is repeated once on a different frequency).

The additional gain of  $S = 3$  is moderate in most cases (up to 25% at position  $C$ ) with an exceptional benefit of 290% at position  $E$ . While the results indicate that  $S = 1$  is a poor choice, selecting either  $S = 2$  or  $S = 3$  inflates to a more delicate trade-off, in which application aspects may play an important role. The main benefit of  $S = 3$  is the increase in PRR. This comes at the cost of a lower data rate (factor 1.5); yet, this number will be smaller in practice when factoring preamble and hardware switching times into the equation. If reliability matters—e.g., in case of control messages or ranging—then using  $S = 3$  is a reasonable and desirable choice. When packet loss can be tolerated—e.g., in certain control tasks—that require high update rates,  $S = 2$  may be favorable.

Results for static gains are very similar and hence omitted. A notable observation is the impact of  $S$  at position  $E$ . Here, we noted a striking increase of PRR from 0% ( $S = 2$ ) to 52% ( $S = 3$ ) at gain level 12 (not shown in the figure). We conclude that in particularly disturbed places (e.g., due to reflections), increasing  $S$  to 3 or possibly even larger values may be required to enable communication at all. Due to the notable influence of position, we believe that an adaptive algorithm may be the method of choice.

## 5.5 Payload Length

Next, we analyzed the influence of payload length on PRR by a series of 200 packets with payload lengths from 0 B to 96 B. In all experiments, we ensured similar pauses between individual packets to achieve comparable experiment conditions in terms of channel cool-down. Results are visualized in Figure 18(b). In general, PRR values w.r.t. positions are similar to those reported previously; i.e., stable and high for positions  $\mathcal{A}$ ,  $\mathcal{C}$ , and  $\mathcal{D}$ , and more fluctuating otherwise. There is no clear trend supporting a strong relation between payload length and PRR for the packet lengths used.

However, the figure reveals dips and notches. For that reason, we explored the percentile of successful synchronizations and SFD detections in Figure 19 to allow for a more detailed understanding.

The results for position  $\mathcal{A}$  in Figure 19(a) indicate that the absolute percentile (PRR) of received packets is smaller for long payloads (of 64 B and more), yet the number of successful synchronizations also drops. Moreover, the percentile of received packets corresponds to the percentile of detected SFDs. This implies that packet loss is not caused by additional payload but deteriorates due to changing environmental conditions. The main finding, hence, is that PRR is maintained over a wide range of payload lengths. However, we want to point out that long packets lead to long transmission times, which may be critical in mobile scenarios. With our default setup, e.g., a

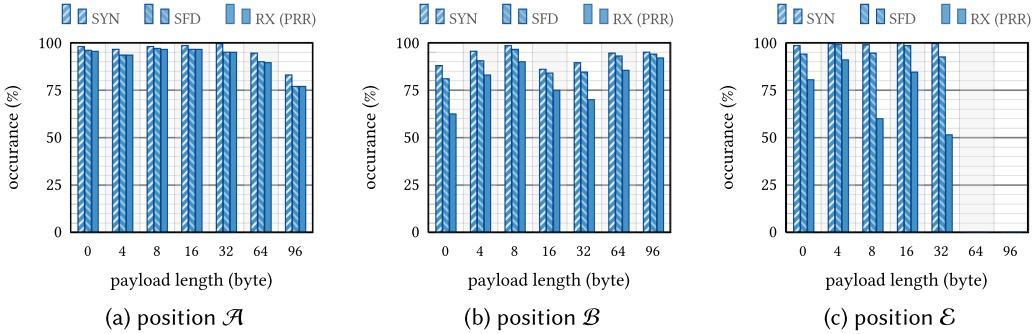


Fig. 19. Statistics of packet reception in terms of synchronizations (SYN), SFD detections, and successfully received packets. For each setup, 200 packets were sent with receivers at different distances (LOS) and payload sizes with automatic gain control enabled. Results at position  $\mathcal{E}$  for 64 B and 96 B are not available due to a depleted laptop battery. Bars labeled RX correspond to PRR.

96 B payload leads to a packet duration of 3.3 s. Even at a moderate  $\mu$ AUV speed of 0.5 m/s, this may cause a 50% symbol shift (or synchronization error, respectively) towards the end of the packet.

Figure 19(b) for position  $\mathcal{B}$  shows more fluctuation regarding all steps of packet reception. Surprisingly, PRR is highest for 96 B payloads, most likely due to better channel conditions. The figure exhibits that a more dominant drop of detected SFD (w.r.t. to synchronizations) leads to a heavier drop of PRR. This indicates poor synchronization, most likely due to reflections, leading to more bit errors and dropped packets. At position  $\mathcal{E}$ , the figures in Figure 19(c) suggest that synchronization and SFD detection stay relatively stable. Apart from this, all previous observations hold, where the effect of low PRR in case of a notable drop from synchronization to SFD detection percentiles is more pronounced.

These findings show that packet length is mainly limited by  $\mu$ AUV speed rather than bit error rates or length-dependent PRR, respectively. In static environments—e.g., stationary sensors [32, 38]—longer packets are advisable to achieve high channel utilization. In mobile scenarios, packet lengths should be chosen based on maximum  $\mu$ AUV speed. The latter is typically within a few meters per second (e.g., Reference [47]). Our experiments support that the most critical step of packet reception is preamble-based synchronization. If it succeeds, then packet reception also succeeds with high probability due to forward error correction, interleaving, and redundancy (cf. Section 5.4).

## 5.6 Preamble-based Synchronization

As precise and correct synchronization—i.e., preamble detection—plays a paramount role for successful packet reception, we investigated the process of synchronization in more detail.

*Preamble Length.* Figure 20 shows the number of continuously detected preamble symbols for the first set of experiments, where a successful synchronization is assumed after 11 consecutive symbols have been detected. At position  $\mathcal{D}$ , for which we observed high PRR values in the previous analyses, the curves in Figure 20(b) are almost flat at 200 detections, the number of sent packets. In all but six cases (all for gain level 0), 200 synchronizations were reported. In fact, 967 packets out of 1,000 packets have been received correctly, where 970 packet starts (i.e., SFDs) were detected.

For positions  $\mathcal{B}$  and  $\mathcal{E}$ , however, many more preamble beginnings (at least one symbol detected) have been observed than packets sent, followed by a massive decay within few symbols. In contrast, only 653 and 237 packets were received correctly, where the majority of receptions was aborted after a missing SFD or excessive bit errors in the packet header, so the payload was

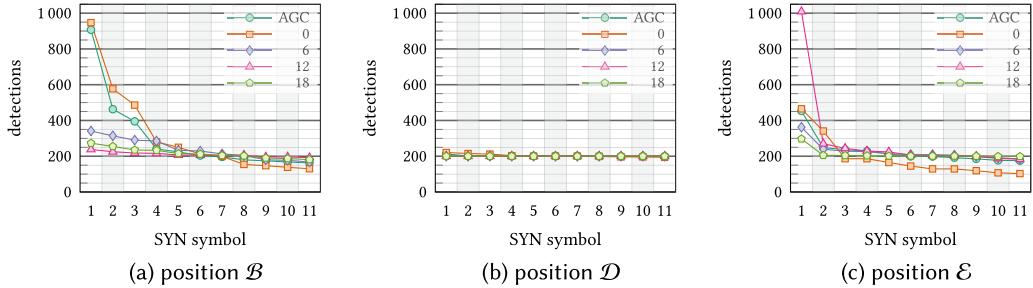


Fig. 20. Statistics of packet reception in terms of synchronizations (SYN), SFD-detections, and successfully received packets. For each setup, 200 packets were sent and with receivers at different distances (LOS) and payload sizes with automatic gain control enabled.

not even investigated. The reason for these divergent numbers is that after an aborted packet reception, the busy channel causes false positive detections of preamble symbols due to frequency cross-talk. The majority of those false positives is eliminated quickly, because the probability of many consecutive false positives is relatively low, as the figures indicate. The quick decay and straightening out of the curves justify our choice of short preambles with a potentially small room for reduction. Longer preambles do not provide any benefit in terms of filtering out those false positives. However, a reduction of preamble length may lead to less precise synchronization, an aspect we did not explore in detail during those experiments. The results from Section 5.5—percentiles of synchronization and SFD detections are similar—yet support a proper choice of preamble length.

*Subsequence Detection.* Preamble detection (cf. Section 3.2) is based on a subsequence detection of the entire preamble. In the default configuration, two subsequences exist. To analyze the effectiveness and usefulness of this design choice, we recorded which subsequence was used for synchronization.

At positions and in scenarios with high PRR, the first subsequence was detected and used in most cases (>99%). Only in conjunction with low PRR, we observed that the second subsequence was useful: 10% at position  $\mathcal{B}$  and 3% at position  $\mathcal{E}$  across all gain levels. A more detailed analysis revealed that at position  $\mathcal{B}$  and gain level 0, 89 packets were received correctly after 90 successful SFD detections. In 18 of these cases (or 20%), the second subsequence was used, meaning that preamble start was not detected at all or the first subsequence has been abandoned within the first cycle. In this particular case, the PRR has been increased by at least 23% (we cannot distinguish if the one failed reception was after synchronization using the first or second subsequence).

Our study conveys the need for advanced preamble detection. It is important to enhance true positive detection to improve PRR. Our results indicate that this is the most critical and determining part of packet reception. On the contrary, false positive detection of entire preambles is observed with very low frequency, so its impact is minor.

## 5.7 Low-power Transmitter

For lower-power applications, we designed a more efficient transmitter with a lower output power, which also comes at a lower cost (cf. Section 3.3). With a dedicated set of experiments, we compared it with its more powerful sibling at positions  $\mathcal{A}$ ,  $\mathcal{C}$ , and  $\mathcal{D}$ . Experiments at different positions were performed sequentially, because we only had a single laptop for receiver logging available. However, we used the same modems as in previous tests and did not change hydrophone positions. Moreover, we only exchanged the transmit board of the sending modem to achieve best possible comparability of results.

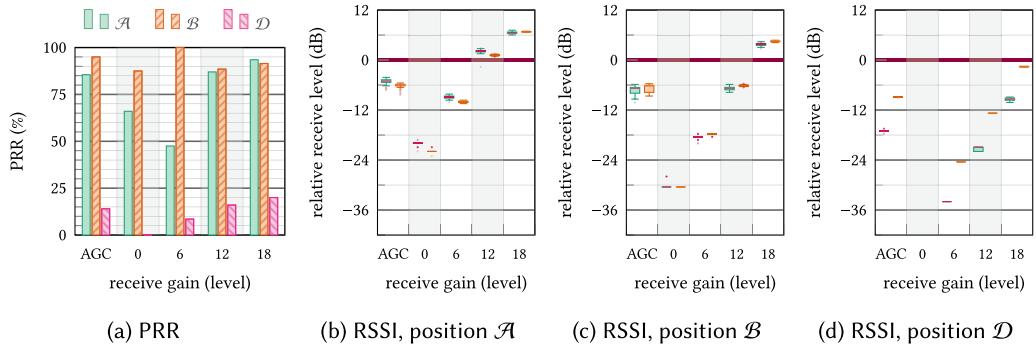


Fig. 21. PRR vs. gain with low-power sender in the default frequency band.

The evaluation, depicted in Figure 21, exhibits fair results for low and intermediate distances but poor PRR at position  $\mathcal{D}$  (ca. 100 m). Surprisingly, results at positions  $\mathcal{A}$  and  $\mathcal{B}$  are flipped compared to previous findings; position  $\mathcal{B}$  now being the better place to receive. Since we did not look into the received signals, we cannot contribute an explanation based on their quality and shape. However, the low-power transmitter has a slightly stronger attenuation of higher frequencies in the communication band (which is not explicitly treated). Therefore, the receive spectrum is expected to differ to some extent, so this is likely one piece of the puzzle.

A look at loudness and RSSI values at the three positions reveals similar behavior w.r.t. Figure 17, with smaller overall numbers for fixed gains. As expected, values for the low-power transmitter are lower by an approximate factor of 2 to 3, corresponding to the factor 2.5 in output voltage. What is notable, though, is the elevated loudness (compared to RSSI) at position  $\mathcal{D}$ , hinting at high noise at the receiver. As there was no obvious change in environmental conditions, we assume this stems from the slightly changed transmit and receive spectrum in combination with frequency-dependent reflections.

The results support that the low-power transmitter is a reliable replacement at short communication distances. Although PRR results at 100 m are low, we expected that communication at this distance should work in many cases. Even if PRR is affected, power consumption is lower by a multiple, so the low-power transmitter is a favorable option for low-power applications, in which data rates are low and retransmissions acceptable.

## 5.8 EU Export-regulations-compliant Frequency Band

As outlined in Section 3.2, the communication frequency band was chosen to achieve high output power (to maximize range) with the AS-1 hydrophone, while keeping both artifacts in the transmitted signal low and preventing high-frequency noise on the receiver side. Both are a consequence of limited sampling frequency. However, the chosen frequency band for communication may not be applicable or legal in certain conditions or regions, e.g., due to potential legislative regulations. One example for such restrictions are EU export regulations, where only the band from 20 kHz to 60 kHz is uncritical, so we repeated parts of our experiments in the range 37.5 kHz to 62.5 kHz, of which only the band from 40 kHz to 60 kHz is used effectively. We go for the upper limit, because the AS-1 has a higher TVR in that area. For this purpose, we modified one receiver to support this frequency band; i.e., shift the bandpass cut-off frequencies. For these tests, we also used our low-power transmitter. The sender  $\mathcal{S}$  sent 100 packets each at three<sup>3</sup> different gain levels and in AGC mode. The receiver was first installed at position  $\mathcal{B}$  and then at position  $\mathcal{D}$ .

<sup>3</sup>The receiver was an older hardware revision with less gain levels.

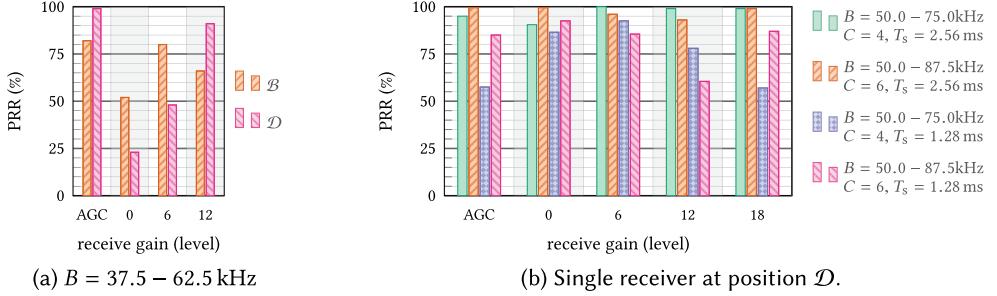


Fig. 22. PRR vs. gain for alternative frequency setups (with adapted equalization) and symbol durations.

The combination of low-power transmitter and lower frequencies reduces output voltage by ca. 10 dB. At position  $\mathcal{B}$ , average RSSI ranges from 3% at gain level 0 to 45% at gain level 12; in AGC mode, average RSSI is 57% at a mean gain level of 13. Values for position  $\mathcal{D}$  are 1% and 13% (for gain levels 0 and 12), and 28% for AGC at an average gain level 16. Loudness to RSSI ratio is again higher at position  $\mathcal{D}$ . These values indicate that the signal is still strong enough at 100 m distance.

However, PRR is lower than with the initial setup in Section 5.3, and it is lower at the farther of the two distances, as shown in Figure 22(a). A closer look at the receive statistics reveals that for position  $\mathcal{B}$ , 77% to 93% of SFDs were detected, yet a large number of bit errors led to corrupted (and hence discarded) packets. An extreme case is observed for gain level 0, where 89% of synchronizations and 83% of SFDs led to only 52% of received packets. At position  $\mathcal{D}$ , however, we observed few bit errors, and the number of received packets almost matches the number of received SFDs. Only in two cases the packet was not received after detection of the SFD (in a total of 263 received SFDs for 100 sent packets).

On a microscopic scale, communication in the lower frequency band at much lower signal level and transmit power works well, particularly in case of AGC. The experiments stress, on macroscope, the location dependency in conjunction with the used frequency band w.r.t. PRR.

## 5.9 Larger Frequency Band and Shorter Symbol Duration

Finally, we investigated communication reliability for elevated data rates by increasing the number of parallel bit transmissions (at the cost of a wider frequency band) and shorter symbols (at the cost of a lower modulation index). In particular, we increased the bandwidth to 37.5 kHz to transmit  $C = 6$  parallel bits (rather than four) without affecting or limiting frequency hopping, and we shrank symbol duration to  $T_s = 1.28 \text{ ms}$ . For the larger bandwidth, we maintained the lower frequency boundary of 50 kHz but moved up the higher boundary to 87.5 kHz. This choice allowed us to expand the frequency band without modifying the receive filter: The larger lowpass attenuation is compensated by the higher hydrophone TVR, and the additional phase delay is not problematic. For each combination of default and modified setup, we sent 200 packets to the receiver at position  $\mathcal{D}$ . Note that the sender was equipped with the high-power transmitter again, so results cannot be compared to those from Section 5.8. With this setup, the maximum data rate is elevated by a factor of 3 ( $1.5 \times 2$ ) to 4,687.5 bit/s, yielding a net data rate of 781.25 bit/s with  $S = 3$  and coding (cf. Section 3.2). PRR values of our experiments are displayed in Figure 22(b) for different gain levels and AGC.

Increasing  $C$  by factor 1.5 for  $T_s = 2.56 \text{ ms}$  does not affect PRR significantly, so this method is an easy way to boost data rate by 50%. The only drawback is the increased bandwidth for communication. However, increasing  $C$  reduces the amplitude of the individual frequency shares (bits) in the signal, hence reducing signal-to-noise ratio (per frequency) and communication range. We

also want to point out that there is a limitation of widening the frequency band due to the transmit characteristics of the hydrophone, as discussed in Section 3.3. Moreover, the processing speed of the microcontroller also puts an upper limit on  $C$ , with not much headroom above a value of 6.

Cutting symbol duration in half has a notably negative influence on PRR, as can be deduced from the figure. This is an expected result in general for the following reasons: Shorter symbols require more accurate preamble synchronization to maintain equal symbol detection quality, yet we expect the contrary for shorter preamble symbols to occur. Even when maintaining preamble symbols with a duration of  $T_s = 2.56$  ms and an unchanged synchronization accuracy, the rate of false (data) symbol detection would increase. Due to the lower modulation index, the former problem is amplified by the resulting fact that cross-correlation of shorter symbols is greater. Moreover, shorter symbols lead to shorter reuse cycles within the hopping sequences, so the benefit of frequency hopping is reduced. In consequence, the impact of reflections increases.

Based on these findings, there is no one-size-fits-all configuration. The ideal choice of parameters depends on the environment and application scenario and may change over time, so, e.g., a rate adaption scheme may improve performance (cf. Reference [42] as an example).

## 5.10 Discussion

In our extensive real-world measurement campaign, we have demonstrated the usability of our modem in a static yet inhospitable (namely, shallow-water) environment. With a fairly simple countermeasure—namely, frequency spreading in conjunction with frequency hopping—acoustic communication based on FSK modulation (with orthogonal frequencies) is improved by multiples compared to existing devices. For a low degree of mobility, we expect similar results for short data packets of up to approximately 100 B at 0.5 m/s of two  $\mu$ AUVs heading in opposite directions. AGC relieves the burden of choosing proper receive gains, and it is generally on par with the corresponding static gain level w.r.t. PRR. Increasing the number of parallel bits is an easy way to elevate data rate. Reducing spreading may be a substitute or additional instrument towards the same goal. Both have a slightly negative impact on PRR only. In contrast, halving symbol duration has been observed to cut PRR by up to 50%, though much less in other scenarios. It is also more sensitive to mobility and hence not recommended in general.

Received signal strength at 150 m distance also gives rise to the assumption that communication range is much larger. Additional experiments are required to give practical evidence, though.

A change of the frequency band can be easily achieved, so the band can be adjusted to legal requirements or to use different hydrophones with our modem. However, any change to the frequency response may lead to significantly altered PRR values at the same position. Using the low-power transmitter impacts PRR slightly, yet reduces transmission cost (consumption) by a multiple. It is, however, already operating outside specification for the default frequency band.

Our findings show that, as with any other wireless technology, pinpointing the actual data rate depends on many aspects, such as position, distance, environmental conditions, and payload size, plus medium access and transport protocol. In contrast to wireless communication, some of these are even more critical; e.g., distance has a massive influence on the achievable data rate due to slow signal propagation. Therefore, we did not analyze the achievable data rate through experiments, because the latter is very application- and scenario-dependent. However, we would like to give realistic examples inspired by Reference [25]. For a unicast communication without or with few acknowledgments—e.g., to transfer compressed images—and a payload size of 100 B, the effective data rate (considering all overhead) will be up to 235 bit/s with the default configuration and 1,039 bit/s for  $T_s = 1.28$  ms,  $S = 2$ , and  $C = 6$ . Data rates scale linearly with PRR. In the same scenario but with per-packet acknowledgments and a distance of 100 m, the data rate would decrease to 235 bit/s and 819 bit/s, respectively. Here, the propagation delay (ca. 67 ms per direction)

puts a hard limit of  $100 \text{ B} / (2 \times 67 \text{ ms}) \approx 6 \text{ kbit/s}$  on the data rate; an Evologics HS modem (rated 62.5 kbit/s) would achieve no more than 5.9 kbit/s. In a broad class of applications (e.g., static monitoring [38] but even mobile tasks [9, 57]), though, data rate is no issue, because data traffic is low and infrequent.

Besides experience with our modem and showing its feasibility in a plethora of applications, the results portray and stress the massively time- and location-dependent characteristic of the (shallow-water) acoustic channel. To achieve unbiased and reliable data (of different configurations), experiments should last for several hours or days and should be interleaving. We observed bursty channel behavior—i.e., communication works well at times with sudden and lasting interruption, a phenomenon known from low-power radio communication and discussed in Reference [1]. Another finding is that due to the frequency selectivity of the channel, there is very narrow room for improving the data rate based on FSK. We conclude a practical limitation of a few kbit/s using FSK. Chirp signals—already employed to some extent by commercial acoustic underwater modems and known from, e.g., radar applications—are an appealing modulation alternative that we have already started exploring from a scientific point of view in Reference [66]. We also found that low-power communication with only a few watts of transmit power at distances of several hundred meters is achievable. From the contrary perspective, this means that for distances of a few ten meters, communication may be possible at much lower power consumption, opening a world of perpetually powered underwater networked sensors.

Finally, our findings support the need for underwater testbeds, which will profit from inexpensive and open hardware like our *ahoi* modem.

## 6 REAL-WORLD RANGING AND MOBILE LOCALIZATION EXPERIMENTS

Due to the promising results from Section 6.2, we went one step further and not only conducted measurements in a mobile setup but also employed a relatively simple localization strategy for a moving  $\mu$ AUV and carried out a test campaign. Here, our goal is to showcase general feasibility of using the *ahoi* modem for ranging and localization so it can be used for research on novel algorithms and their real-world evaluation. Subsequently, we explain our approach, test environment, and results.

### 6.1 Experiment Setup

$\mu$ AUV *Integration*. To enable and evaluate self-localization in a mobile scenario, we equipped a HippoCampus  $\mu$ AUV provided by the authors of Reference [30] with the *ahoi* modem, as shown in Figure 23. HippoCampus perfectly fits our targeted use case of low-cost, small  $\mu$ AUVs, and it can be easily extended for various research purposes. We attached HippoCampus via an iron rod to an aluminum cross with blocks of polystyrene at its ends—floating on the surface—to keep the robot at a constant depth: 80 cm, in our case. The reasons for this provision were to:

- keep the robot at a fixed depth despite the lack of a functional depth control at time of testing,
- add a GNSS module at the surface to obtain reference position data (ground truth),
- allow for robot RC control to sail trajectories, and
- observe position and orientation of the submerged robot at far distances and despite virtually no visibility at depths beyond a few ten centimeters.

For simplicity and easy live-tracking of the evaluation process, we integrated a Raspberry Pi Zero W into the robot. An external Wi-Fi antenna was attached to the Raspberry Pi. The antenna was fixed at the top of the aluminum cross to ensure continuous connection between the Raspberry

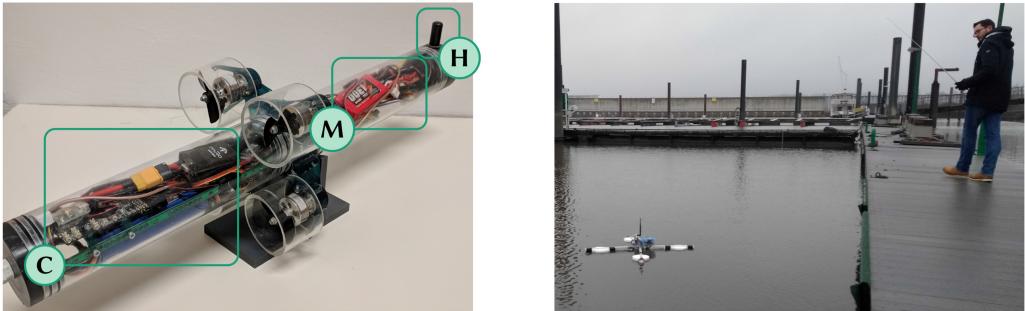


Fig. 23. Left: HippoCampus  $\mu$ AUV with integrated (C) controller and main battery; (M) *ahoi* acoustic modem and Raspberry Pi, and (H) hydrophone mounted to the hull of the  $\mu$ AUV. Right: Submerged HippoCampus (not visible) in the marina of TuS Finkenwerder with external Wi-Fi antenna and GNSS receiver mounted on the cross carrier.

Pi and the rest of our evaluation equipment. The *ahoi* modem was connected to the Raspberry Pi via serial interface. The hydrophone used for transmission and reception of the packets was directly mounted to the hull of the  $\mu$ AUV. To compare our localization results against a ground truth, we connected a Navilock NL-8001U GNSS receiver to another Raspberry Pi at the water surface. During our evaluation, we controlled the robot with an off-the-shelf remote control.

*Localization Algorithm.* For localization of the  $\mu$ AUV, we used a round-based concept with four fixed anchors at positions  $\mathcal{A}$  to  $\mathcal{D}$ . The concept is similar (and compatible) to Reference [9] (cf. Section 2.5), but improves latency by leveraging a single packet to collect distances from all anchors. Medium access is scheduled (TDMA) so collisions are completely prevented. At the beginning of each round, the  $\mu$ AUV sent an empty<sup>4</sup> ranging request as broadcast. Upon reception of a request, the anchors replied with a ranging response consecutively: Anchor  $\mathcal{A}$  replied immediately, anchors  $\mathcal{B}$  through  $\mathcal{D}$  replied with a single, double, and triple delay, respectively. Delays and round length were chosen as follows to prevent collisions and allow for a bit of extra time to let echos on the channel fade. In our setup, packet duration for request and response (4 B payload) is 292 ms and 476 ms, respectively. At distances of at most 75 m between anchors and  $\mu$ AUV, propagation delay is below 55 ms. To avoid collisions, the delay between anchor responses needs to encompass (at least) response duration plus double propagation delay. We hence chose a delay of 750 ms. Each round has to encompass one response per anchor of duration up to the delay period plus request duration, leading to a round interval of 4 s. Figure 24 shows an example round with the chosen values.

As a side note, we want to point out that these numbers are conservative to study general feasibility of mobile communication and localization. However, we are aware that large delays result in ranging errors due to movement of the  $\mu$ AUV, which we will study in follow-up work.

During our experiments, the  $\mu$ AUV recorded all received ranging responses, and we performed localization offline through lateration (using least-squares regression) at the end of each round with at least three received ranging responses (i.e., three or more updated distances). Technically, this approach demands that anchor positions are known by the  $\mu$ AUV for simplicity. In a known environment, such information could be pre-installed on the  $\mu$ AUV or acquired from a control center before start of a mission. In a scenario with mobile anchors as in Reference [9], this information could be included in the responses.

<sup>4</sup>This request is an ordinary data packet and can be leveraged to transmit arbitrary information to the anchors.

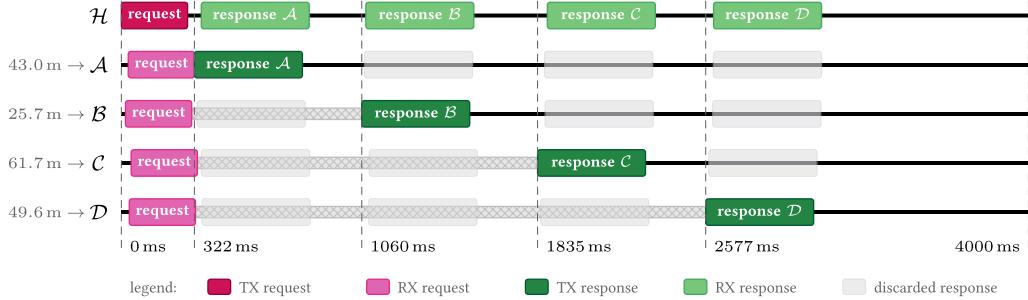


Fig. 24. Example of round-based localization for packet-based ranging with values and positions according to our experiment setup. The position of the HippoCampus  $\mu$ AUV ( $\mathcal{H}$ ) is at the farthest point of all experiment runs, i.e., the final position of run 4 at  $(-37 \text{ m}, -22 \text{ m})$ ; cf. Figure 25(a) and Figure 28(b). Distances between  $\mu$ AUV and anchors are displayed left of anchor names ( $\mathcal{A}$  to  $\mathcal{D}$ ). Dashed vertical lines with times indicate start times (relative to round begin) of sent responses.

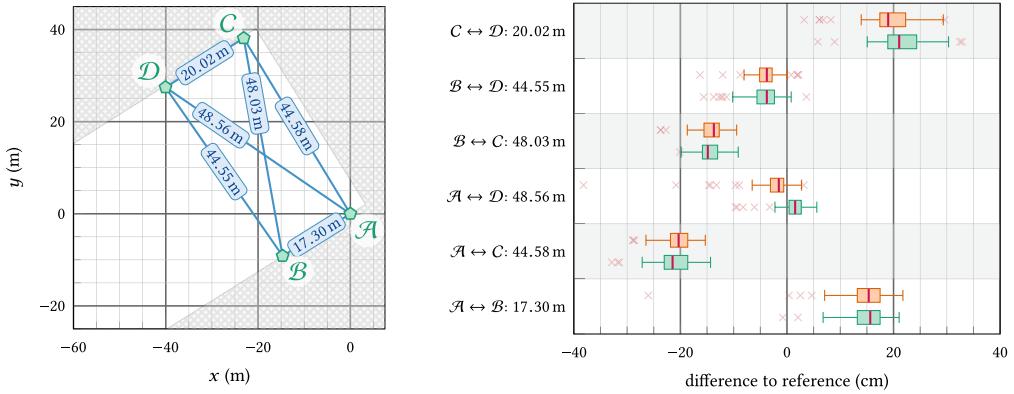
The employed algorithm has the benefit over querying all anchors individually that overall round interval (and therefore localization intervals) can be reduced, because only a single request is sent. A drawback is that delays have to be configured appropriately. Another advantage evolves in combination with the final localization step. Because ranging requests are sent simultaneously to all anchors, the first half of distance measurements is accurate and for the same point in time. Only for the responses, there is an error due to  $\mu$ AUV movement. If an individual request were sent for each anchor, then both parts of ranging would be error-afflicted, unless  $\mu$ AUV speed and heading were known, so compensation would be possible. In our case, the ranging error is at most 3 cm for the first and 26 cm for the last anchor. These figures are calculated by multiplying (average)  $\mu$ AUV speed with the time until the response is sent (which embraces duration of the request plus propagation delay and anchor response delay). Due to application of two-way time-of-flight ranging and the simultaneous request, errors are half of calculated values.

*Anchor Setup.* We installed the four anchors in a near rectangular shape at the site from Section 5.1 with hydrophone depths of 1.7 m. The setup is visualized in Figure 25(a). We used plastic tubes to achieve an almost perpendicular arrangement of hydrophones w.r.t. the surface. We conducted our experiments on January 7, 2019, from 11 am to 4 pm. The water temperature was  $4.9^\circ\text{C}$ , and we noted steady but little surface waves. Salinity was 0.51 ‰, resulting in a speed of sound of  $c = 1426.4 \text{ m/s}$ . We determined precise global anchor positions with a system provided by Prof. Andree from the Institute of Solid Construction at TUHH. It combines a high-precision JAVAD Triumph-2 GNSS<sup>5</sup> with post-processing multiple raw data streams using differential positioning information.<sup>6</sup> Typically, accuracy is within few centimeters at most. For better illustration, all coordinates were transformed to UTM format.

*Mobile Experiment.* Finally, we ran four experiments with the prepared HippoCampus  $\mu$ AUV sailing various RC-controlled trajectories. All modems used the default configuration as described in Section 3.2 with AGC enabled. Run 1 lasted 17.5 min (300 ranging requests with a round interval of 3.5 s due to a harmless misconfiguration); all other runs lasted 13.3 min (200 ranging requests with a round interval of 4 s). For run 2, we only considered the first 79 rounds, because a water leak caused a short in the robot and interrupted communication between robot and modem (no more ranging requests were sent). In all runs, the  $\mu$ AUV was moving at an average speed of 0.2 m/s

<sup>5</sup><https://www.javad.com/jgnss/products/receivers/triumph-2.html>.

<sup>6</sup>[https://intranet.tuhh.de/aktuell/pressemitteilung\\_einzeln.php?Lang=en&id=11669](https://intranet.tuhh.de/aktuell/pressemitteilung_einzeln.php?Lang=en&id=11669).



(a) Anchor setup; only white area is accessible.

(b) Ranging results. Single value  $-64$  cm for  $\mathcal{D} \rightarrow \mathcal{B}$  not shown.

Fig. 25. (a) Anchor setup with GNSS-based UTM positions relative to anchor  $\mathcal{A}$ . (b) Comparison of GNSS distances and ranging results with the *ahoi* modem. Differently colored boxes indicate different communication/ranging directions. Number of successful rangings is between 92 and 100.

according to GNSS data. An analysis of the communication reliability is carried out in Section 6.3; results of the mobile localization are presented in Section 6.4.

## 6.2 Ranging Accuracy

Before looking at the mobile case, we studied the ranging accuracy of the *ahoi* modem by means of 100 ranging requests per direction for each pair of anchors. This analysis also is a mandatory prerequisite for the validity of (mobile) localization, because the latter is only feasible if ranging in the static case is reliable and sufficiently precise.

In our experiment composed of 1,200 requests, 1,178 responses were received. The minimum per-link result was 92 responses. Figure 25(b) shows the GNSS reference distances of the anchors and the acoustic measurements through a box plot with quartiles. The results show a very narrow spread and only a few outliers. 50% of all values around the respective median are within 6 cm (equivalent to only  $42\ \mu\text{s}$  or 8.4 samples synchronization deviation; cf. Section 5.6) and the maximum variation is 60 cm—the maximum theoretic error would be 3.65 m. Results are consistent w.r.t. measurement directions. The deviation from GNSS-based distances is below 65 cm in all cases, and below 25 cm in more than 95% of cases. We assume that the static error of underwater distance measurements w.r.t. GNSS distances is mainly caused by horizontal displacement due to imperfect mounting and positioning of the hydrophones underwater; an error of  $1^\circ$  already causes a displacement of 3 cm in our setup. Reflections at the water surface, small hydrophone movements due to water current, and variation of packet-based synchronization add to this error and introduce the visible deviation of results. However, we argue that the achieved precision is sufficient in most real-world deployments with standard GNSS receivers with tens of centimeters or even multi-meter localization error.

## 6.3 Results of Mobile Communication

*Ranging Success (Packet Reception Rate).* Figure 26 shows PRRs of ranging requests (at the anchors) and responses (at the moving  $\mu$ AUV) for all experiment runs. The figures reveal that PRR is larger for the direction from the moving robot to the stationary anchors than in the opposite direction. This observation hints at possible noise due to dynamic pressure at the hydrophone during robot movement, bubbles, or other dynamic effects. A closer look at timely behavior did not reveal

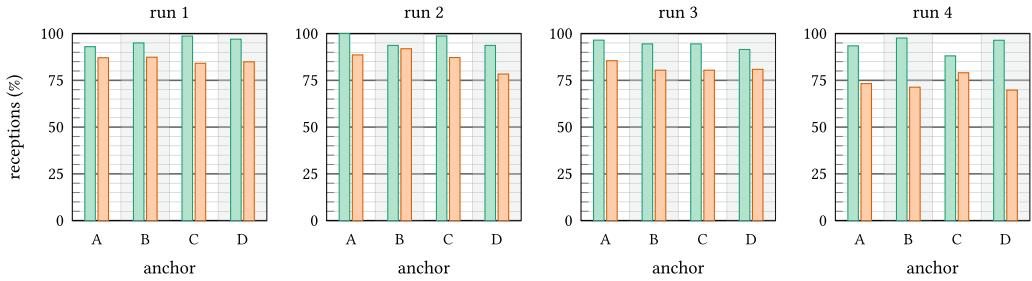


Fig. 26. PRR of ranging requests (from  $\mu$ AUV at anchors) and responses (at the  $\mu$ AUV from anchors). Note that PRR of responses is relative to received requests.

Table 1. Packet Reception and Localization Frequency Statistics of Experiment Runs with Mobile  $\mu$ AUV

(a) Number of received responses per anchor. (b) Number of responses to a ranging request per round.

Anchor	Received responses (%)			
	Run 1	Run 2	Run 3	Run 4
A	81.0	88.6	82.5	68.5
B	83.0	86.1	76.0	69.6
C	83.0	86.1	76.0	69.6
D	82.3	73.4	74.0	67.3

Responses per round	Relative frequency (%)			
	Run 1	Run 2	Run 3	Run 4
0	0.7	1.3	1.5	2.4
1	2.3	0.0	6.0	9.5
2	14.7	10.3	15.0	31.5
3	33.0	34.6	36.5	36.3
4	49.3	53.8	41.0	20.2

Note that percentile of received responses may differ from PRR, as explained in the text.

notable location or heading dependency. In runs 1 to 3, request PRR was 79% to 94%. Only in run 4, this number drops to 70% to 80%, which is mainly caused by a short period of virtually no reception at all (around round 150; cf. Figure 28(b)). This period coincides with the  $\mu$ AUV close to a metal pillar at map position ( $-55$  m,  $20$  m) in Figure 25(a). Therefore, and in combination with experiences from preliminary tests, we assume that this pillar caused massive reflections, leading to severe packet loss. As we will show subsequently, the overall percentile of received ranging responses is sufficient for reliable self-localization. However, the considerable degradation of PRR at a mobile receiver requires thorough analysis and remedies, which are currently under investigation and beyond the scope of this study.

Since localization depends on reception of ranging responses (per round) of at least three anchors, we dug deeper. Table 1(a) shows the percentiles of received ranging responses vs. sent requests, which are slightly smaller than response PRRs, yet above 67%. These numbers stress the need to find the reasons for loss in the direction of the  $\mu$ AUV, because this general issue will most likely be found when using other modems as well. Yet, we find that acoustic localization with four anchors is feasible, since the additional anchor provides redundancy. Table 1(b) exhibits that between 56.5% (run 4) and 88.4% (run 2) of all rounds, at least three anchors were available and localization hence possible. These numbers rise to almost 90% when also considering the cases of only two received responses per round. In many cases, results from two anchors may be sufficient for localization by excluding one of the two solutions of the localization operation through plausibility checks (e.g., by checking the traveled distance or  $\mu$ AUV speed and heading, respectively). Due to no unexpected findings, we omit results regarding PRR vs. position (relative to anchors).

*Signal Receive Level and RSSI.* Complementing the findings in Section 5.3, we explored received signal level, gain, and PRR in the mobile case. Figure 27 shows RSSI and gain of all successful

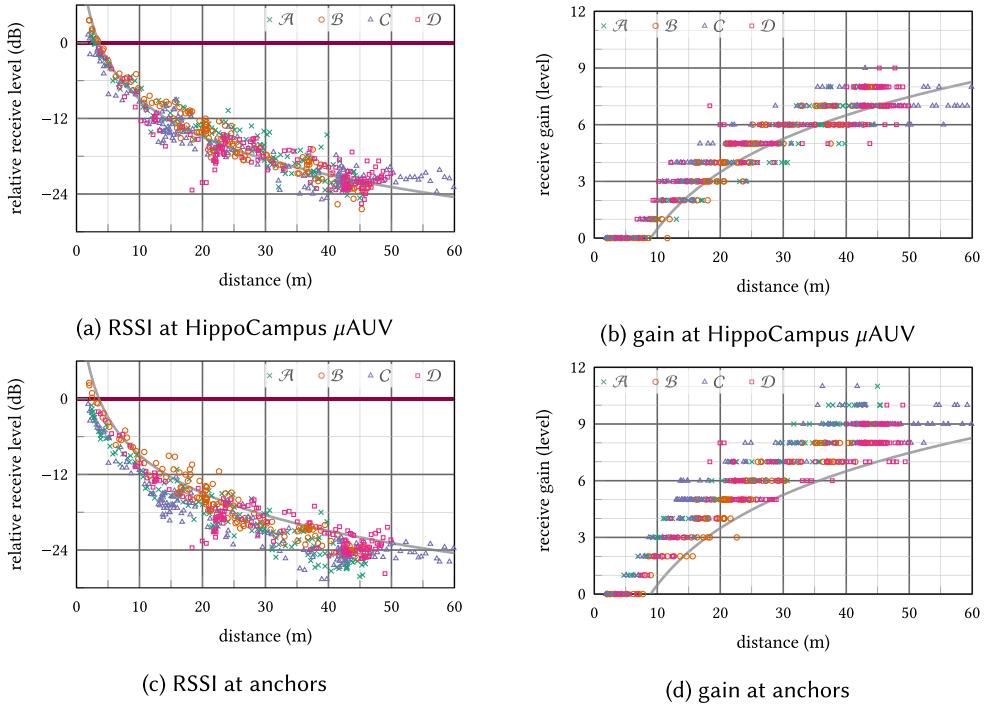


Fig. 27. Relative RSSI (left, cf. Figure 17) and receive gain of successfully received packets by HippoCampus (upper row) and anchors during run 3. Distances on the x-axis are based on acoustic ranging. Solid gray lines indicate theoretic behavior according to path loss in Equation (1) and Equation (2).

rangings during run 3 for the HippoCampus  $\mu$ AUV and all anchors. Signal loudness is similar to RSSI and behavior already discussed in Section 5.3, so we omitted the plot. Results across all four runs are comparable.

The figures reveal a notable and expected relation of distance vs. RSSI and gain, respectively. This is indicated by the ideal course—linear increase (gain) and inverse-linear decay (RSSI), respectively—in the plots. RSSI (Figure 27(a)) and gain (Figure 27(b)) are homogeneously distributed across all anchors, and variation for the individual modems is due to the changing relative position of the  $\mu$ AUV. Particularly for anchor C, there are multiple  $\mu$ AUV positions with equal distance to the anchor, leading to a wide variation of RSSI and gain levels. In general, these findings also apply to the packets received at the anchors (Figure 27(c) and Figure 27(d)). However, RSSI is 2 dB to 4 dB lower and gain one to two levels higher. Particularly at short distances, this may be caused by  $\mu$ AUV movement (relative to the anchor) leading to changing multi-path scenarios. The static distance error (cf. Section 6.2) and the error due to the ranging delay (cf. Section 6.4) may also contribute to a shift (on the x-axis in the plots). Another potential reason for this observation could be the fact that the hydrophone is mounted to the  $\mu$ AUV casing, leading to signal attenuation during transmission (from the  $\mu$ AUVs).

With an average gain level of 9 out of 18 at a distance of 50 m, the results support a considerable head room of 18 dB or almost a factor 10, respectively, for communication range. A different interpretation would be that transmit power could be considerably reduced for distances up to 100 m. Another insight worth noting is that ranging based on RSSI would introduce large errors (several meters, in most cases), so the employed TWR is preferable.

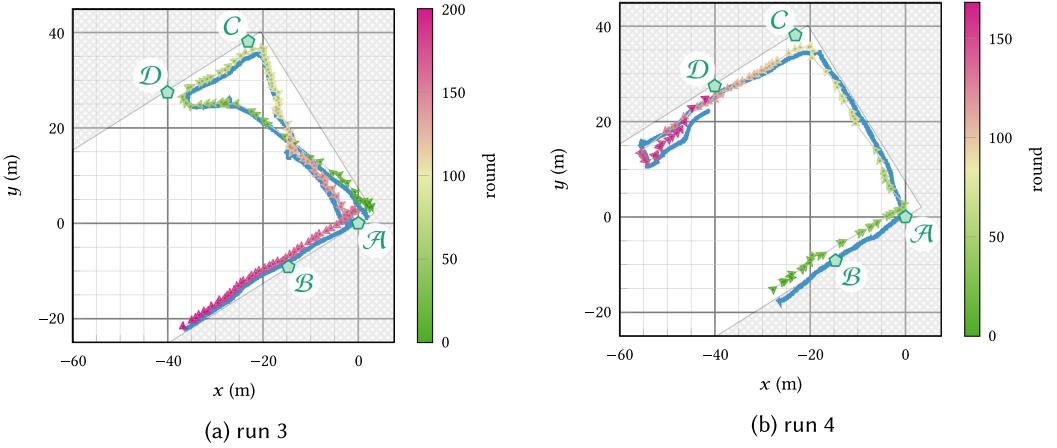


Fig. 28. Comparison of GNSS tracks (fine-grained blue) and estimated trajectory (shaded triangles pointing in the direction of movement) based on acoustic ranging and acoustic localization. Gray cross-hatched areas show non-accessible areas (due to jetties; cf. Figure 25(a)). Positions are based on UTM coordinates relative to anchor  $\mathcal{A}$ .

#### 6.4 Results of Mobile Localization

*Trace Analysis.* In all runs, there is a clear match between visualized GNSS and acoustic localization trace. Figure 28 exemplarily shows the trace for runs 3 and 4. The results for the third run in Figure 28(a) exhibit a dense mesh of localization points with few dead spots. The difference between GNSS track and acoustic localization is very low, within 2 m in most cases. Results of run four in Figure 28(b) indicate a higher rate of missing points and larger difference between GNSS and acoustic localization. Mean gaps (distances) between GNSS and localization positions range from 1.4 m (run 3) to 2.6 m (run 1) with a standard deviation from 0.4 m (run 3) to 1.3 m (run 1). The reasons for these are manifold, so we did evaluated position and distance errors in more detail.

First, GNSS positions of the moving  $\mu$ AUV are error-afflicted themselves and—unless determined through a high-precision device that requires minutes without movement to obtain a precise result (and is hence not usable)—do not suffice as ground truth to quantify errors. It is evident from Figure 28(b) that mobile GNSS positions up to round 50 are off by at least a meter, because the GNSS track is on the jetty. Here, we want to point out that we confirmed jetty positions with the high-precision GNSS system explained in Section 6.1, and that we used those measurements to draw jetties in the figures. Moreover, we always kept a minimum distance of 1 m between  $\mu$ AUV and jetties. Therefore, GNSS positions determined by the  $\mu$ AUV can be identified as being imprecise. However, they are valid in our case (no walls and obstacles that potentially cause massive GNSS errors) for assessing qualitatively whether acoustic localization produces reasonable results.

Second, the employed acoustic localization is rather primitive and aims at showcasing the general feasibility of localization with the *ahoi* modem and gathering first real-world experience for the design of more advanced localization algorithms. There is, e.g., no compensation of  $\mu$ AUV movement and orientation. The former could be addressed with filtering or speed estimation. In our experiment, though, the error due to mobility is expected to be small (cf. Section 6.1). Regarding the latter, there is a 28 cm distance between hydrophone and center of the  $\mu$ AUV (where the GNSS receiver is positioned). This displacement has to be factored into the equation for localization and its evaluation.

*Analysis of Localization Gap.* However, we inspected gaps between GNSS and acoustic localization positions along the track of run 4. The corresponding figures and behavior are visualized in Figure 29. Under the assumption of piece-wise coherent GNSS data—meaning that relative positions are correct in short time frames or while the heading of the  $\mu$ AUV is not changed—we can make the following observations: In the first 50 rounds, when the  $\mu$ AUV sails from start to anchor  $\mathcal{A}$ , the gap for the distance to anchor  $\mathcal{B}$  jumps up once anchor  $\mathcal{B}$  is passed. The same behavior is seen when sailing towards and away from anchors  $\mathcal{A}$  and  $\mathcal{C}$ . This indicates the error due to the offset of the hydrophone (w.r.t. GNSS receiver position on the aluminum cross). The figure also shows an even distribution of anchors used for localization, and there are no heavy jumps in distance and gap plots when anchors are changed in subsequent localization steps. This supports the general feasibility of the algorithm (for low  $\mu$ AUV speeds), particularly implying that speed compensation may not be required. Moreover, the figure stresses the large absolute gap (of up to 4 m) in the first half of the run due to the offset between GNSS and acoustic tracks. The latter determines overall gap. Conclusions are twofold. First, the results clearly indicate that a better ground truth or different method is required to assess mobile acoustic localization in a quantifiable manner. Second, acoustic localization results are consistent and gaps w.r.t. GNSS-based localization are due to an offset of the tracks and not due to measurement noise and (random) jumps.

## 6.5 Discussion and Limitations

During our practical evaluation of ranging and localization, we faced several challenges. The main and most important one is the determination of an accurate ground truth for mobile localization but also for the static case.

Even in static setups, it is difficult to align hydrophones (of anchors) exactly below the surface reference point. In many cases, jetties are mounted floating to cater for tides and other environmentally caused shifts of water level. Small movements at the surface and tilting has a considerable influence on submerged hydrophones. Even if accurate position knowledge at the surface can be obtained via high-precision GNSS devices, tilt and orientation sensors are needed. While in practice, errors of several centimeters may be tolerable, thorough assessment of ranging precision and accuracy becomes troublesome. One remedy would be to obtain a ground truth through wide-band chirp signals [66], which require precise (micro-second) time synchronization of sender and receiver. In most scenarios (and on longer distances beyond a few ten meters), tape is not feasible.

These observations carry over to the mobile case. Obtaining absolute position errors of a mobile system with acoustic localization is extremely difficult, even with the high-quality, high-precision equipment that was available. The main issue is imprecise GNSS positioning information of the mobile device, which is required to assess acoustic localization.

However, we were able to show for the static case that acoustic ranging has low variation and is within hardly avoidable uncertainty boundaries. We hence argue that it is precise enough for stationary and mobile acoustic ranging. With the suggested improvements, accuracy could yet be improved, if required. Moreover, acoustic underwater localization with the *ahoi* modem (and also similar devices) is feasible. An evaluation based on a relatively simple method has produced reasonable, traceable, and smooth track results. Due to the long delay between reception of multiple ranging responses and the delay between ranging request and responses, movement of the  $\mu$ AUVs is limited to low speeds at the moment. With more sophisticated algorithms (e.g., including speed and heading compensation), high-quality underwater localization with inexpensive devices is enabled for  $\mu$ AUVs even at higher speeds.

Our analysis also reveals that the assumption made by many  $\mu$ AUV developing researchers—that the position of a  $\mu$ AUV equipped with a simple GNSS receiver can be assumed to be known—is at least questionable. Particularly in small-scale swarms—with distances of few meters between

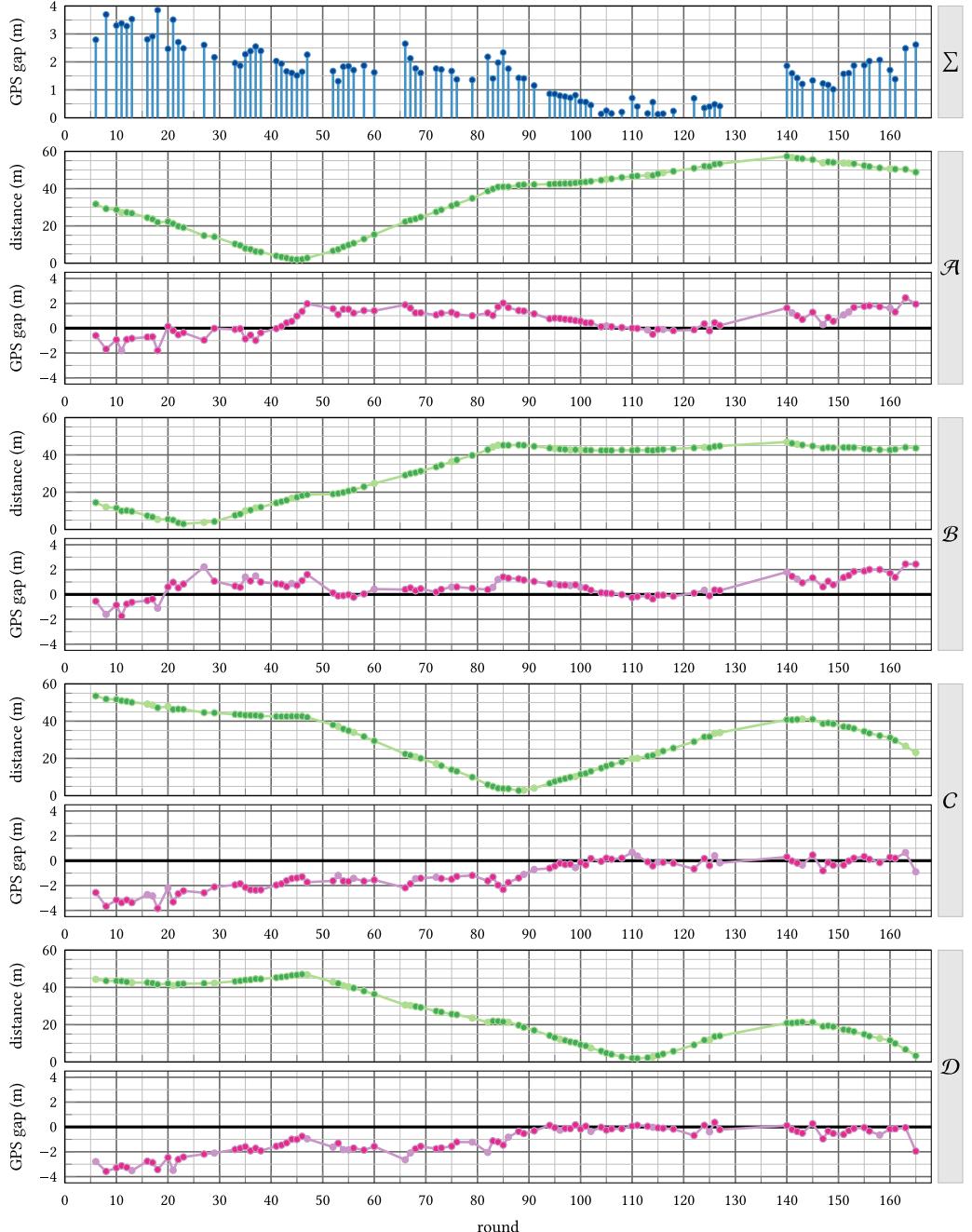


Fig. 29. Gap between GNSS and acoustic localization positions for the track shown in Figure 28(b) (top plot labeled  $\Sigma$ ). Distances between  $\mu$ AUV and anchors determined by acoustic ranging (green, top of each group) and gap between ranging and GNSS-inferred distances for all anchors (positive values imply that ranging is larger than GNSS distance). Light-colored marks are displayed for all localization positions, dark-colored marks only when the corresponding anchor was used for localization; i.e., a ranging response was received.

devices—relative positioning with surfaced robots may not be reliable with inexpensive GNSS receivers, which introduce position errors of several meters already.

Finally, our experiments demonstrate that communication and localization of  $\mu$ AUVs with an inexpensive, low-power acoustic device like the *ahoi* modem is feasible and works well. However, environmental conditions—e.g., pillars causing reflections—may hamper connectivity due to inter- and intra-symbol interference. A recent study on chirp modulation in Reference [66] indicates that mitigation will be available in the future.

## 7 CONCLUSION

Underwater sensor networks in general and micro autonomous underwater vehicles ( $\mu$ AUVs) in particular are highly active and rapidly advancing research topics. However, research progress and real-world application is dependent on devices for and practical experiences with inexpensive and low-power communication. An important driver is the longtime neglected domain of shallow waters with special demands on small devices and low prices plus additional challenges regarding the communication channel with massive reflections and reverberation. Unfortunately, commercial devices are typically prohibitively expensive, excessively large, or closed source. Devices developed in research projects are—in many cases—no longer maintained, need special hardware and knowledge, or are inaccessible. In summary, performing research with acoustic underwater communication, which involves practical field tests to match theory and praxis, is extremely difficult to achieve.

For that reason, we developed a small, low-power, inexpensive open-source acoustic modem called *ahoi*. It is based on standard, off-the-shelf components, and it can be built by anyone capable of handling a soldering iron and reading a circuit diagram. We conducted lab experiments to demonstrate its low power consumption and that its analog characteristics meet the design goals.

Moreover, we conducted and evaluated extensive real-world experiments to show the function of our modem in praxis and to gain general insights into underwater communication, ranging, and localization. The article goes beyond existing research and entails results from stationary and mobile setups, including an evaluation of a mobile underwater localization experiment conducted in a marina of the river Elbe.

With a lightweight FSK-based communication stack, the *ahoi* modem achieves ranges of more than 150 m, potentially being capable of a multiple of this. We explored the influence of symbol duration, communication bandwidth, and the benefit of a spreading technique to counter reflections (mainly from the water surface). We identified preamble-based synchronization as a crucial step in the reception process, explored details of our implementation, and discussed open issues and improvements. We also verified the linear relationship between signal strength (amplitude) and distance and experienced a burstiness of the channel comparable to wireless radio sensor networks. We found that PRRs of 75% to 100% are achievable; only in the vicinity of heavy reflectors, a severe degradation is observable. We investigated the influence of packet length and found that—due to relatively heavy coding and redundancy—its influence is low for payloads from 0 B to 96 B.

Moreover, we integrated the modem into the HippoCampus  $\mu$ AUV and studied PRR, ranging, and localization with the *ahoi* modem in a mobile experiment. Our experiments support the general feasibility of underwater communication with a mobile robot and its ability to achieve proper self-localization with four stationary anchors despite a relatively simple localization algorithm. Additionally, we identified relevant issues with current assumptions of robot swarms that, e.g., mandate precise surface localization with GNSS.

Our results can be transferred and are applicable to similar devices and robots. Because hard- and software of the modem is open source, we invite and enable researchers and industry to use our platform and develop advanced and alternative communication architectures and methods.

However, the present implementation already achieves reliable communication in many scenarios, including mobility, and can hence be used to carry out research regarding localization, networking (e.g., routing), and application-level topics. At the time of this writing, we have shipped five devices to the University of Lübeck for integration in MONSUN, and sent six devices to a company in North America for testing. Moreover, there is a local research cooperation at TUHH and one with IMTEK at the University of Freiburg, Germany.

Currently, we are investigating the benefit of chirp spread spectrum and lightweight implementations (for the *ahoi* modem and similar devices) to improve reliability and data rate. We are looking into localization and route-planning algorithms based on acoustic ranging for  $\mu$ AUVs. One of our next steps is the integration of the *ahoi* modem into a BlueROV. Finally, we have hybrid networks with stationary surface buoys and autarkic  $\mu$ AUVs on our agenda.

## APPENDIX

### A COMPARISON TABLE OF ACOUSTIC MODEMS

The following table contains an in-depth comparison of available acoustic modems. All values have been obtained with the highest care but are subject to change and are supplied without guarantee. In case of commercial devices, please refer to the latest datasheet or website. Due to availability, weights and dimensions for research devices are provided without casing and transducer; values for commercial devices are provided as per datasheet. Weights are in air.

name	reference	price (€)	availability / distribution	firmware	processing	network size	dimension (mm)	weight (g)	housing
A ahoi	ca. 600	open source / can be ordered	open source	$\mu$ C	255	50 × 50 × 25	150	yes (100 m) <sup>d</sup>	
<b>research devices</b>									
R1 WHOI Micromodem 2.0	[27, 28]	ca. 8,000	open source / commercial	open source	DSP	n.a.	79 × 127 × 44	n.a.	
R2 UWModem	[10]	250	open source	open source	FPGA	n.a.	n.a.	n.a.	
R3 n.a.	[50]	n.a.	software based	n.a.	FPGA	n.a.	n.a.	n.a.	
R4 Itaca modem	[70]	n.a.	n.a.	n.a.	$\mu$ C	n.a.	n.a.	n.a.	
R5 n.a.	[35]	n.a.	breadboard design	n.a.	Arduino + $\mu$ C	n.a.	n.a.	n.a.	
R6 n.a.	[90]	n.a.	n.a.	n.a.	RaspPi + $\mu$ C	n.a.	>85 × 56 × 20	n.a.	
R7 Nanomodem	[49]	60	n.a.	n.a.	n.a.	n.a.	Ø 42 × 60	n.a.	
R8 n.a.	[71]	n.a.	n.a.	n.a.	$\mu$ C	n.a.	Ø 80 × 100	n.a.	
R9 n.a.	[18]	>3,000	software based	n.a.	PC + FPGA	n.a.	n.a.	n.a.	
<b>commercial devices</b>									
C1 Eulogy SEC 18/34	[24]	ca. 8,000	commercial	proprietary	FPGA	n.a.	Ø 110 × 265 (170 <sup>e</sup> )	2,445	yes (200 m) <sup>b</sup>
C2 Eulogy SEC M1HS	[24]	n.a.	commercial	proprietary	FPGA	n.a.	Ø 63 × 310 (235 <sup>e</sup> )	1,120	yes (200 m) <sup>b</sup>
C3 Teledyne ATM-903	[72]	n.a.	commercial	proprietary	n.a.	n.a.	Ø 75 × 420	3,200	yes (500 m) <sup>b</sup>
C4 Sonardyne Modem 6 Sub-Mini	[65]	ca. 15,000	commercial	proprietary	DSP	15	130 × 60 × 25	530	yes (1,000 m) <sup>b</sup>
C5 Aplicon SeaModem	[16]	n.a.	commercial	proprietary	n.a.	n.a.	Ø 56 × 79	235	yes (750 m) <sup>b</sup>
C6 Tritech Micro Data Modem	[86]	n.a.	commercial	proprietary	DSP	15	Ø 55 × 106	676	yes (2,000 m) <sup>b</sup>
C7 blueprint subsea X110	[11]	n.a.	commercial	proprietary	DSP	15	Ø 85 × 600	n.a.	yes (6,000 m) <sup>b</sup>
C8 devologic HAMBASE	[19]	n.a.	commercial	proprietary	DSP	64	100 × 80 × 20	n.a.	—
C9 DSPComm Aquacomms Gen2	[22]	n.a.	commercial	proprietary					
modulation	freq. band (kHz)	bandwidth (kHz)	data rate <sup>c</sup> (bit/s)	SPL (dB re 1 $\mu$ Pa/m)	tx (W)	rx (mW)	sleep (mW)	range (m)	notes
A FSK, OFDM	25–87.5	up to 37.5	260–4,700	up to 161	<5	300	<1	>150	
<b>research devices</b>									
R1 FSK, PSK	various	various	80–5,400	n.a.	35–150	620–2,550	<2	1,500	
R2 FSK	32–38	6	200	up to 192	2–40	375–750	n.a.	2,000	custom piezo
R3 BPSK	100–1,000	n.a.	80,000	n.a.	n.a.	n.a.	n.a.	50	software defined radio
R4 FSK	85	n.a.	1,000	n.a.	0.04–0.13	24	0.003	100	
R5 FSK	40–43	3	1,200	n.a.	n.a.	n.a.	n.a.	<2	
R6 FSK	30.5–32.5	2	100–1,200	n.a.	n.a.	n.a.	n.a.	1	custom transducer
R7 CSS	24–28	4	40	168	0.5	<10	n.a.	2,000	
R8 DSSS (BPSSK)	11–14	3	55	n.a.	n.a.	n.a.	n.a.	200	
R9 OFDM, CSS	88–112	24	up to 30,000	n.a.	n.a.	n.a.	n.a.	200	software defined radio
<b>commercial devices</b>									
C1 S2C, PSK	18–34	16	up to 13,900	n.a.	2.8–80	<800	2.5	3,500	
C2 S2C, PSK	120–180	60	up to 62,500	n.a.	3.5–10	800	0.5	300	
C3 MFSK, PSK	9–27	5	80–15,360	n.a.	20 <sup>a</sup>	768 <sup>a</sup>	17 <sup>a</sup>	2,000–6,000	
C4 n.a.	21–32.5	11.5	200–9,000	187	n.a.	n.a.	n.a.	3,000–4,000	
C5 FSK	25–35	10	750–2,250	n.a.	up to 40	n.a.	n.a.	400	
C6 n.a.	20–28	8	40	169	7.9	700	n.a.	500	
C7 n.a.	24–32	8	100	172	<10	500	n.a.	1,000	
C8 MFSK, OFDM, MDPSK	40–65	25	up to 10,000	190	up to 10	n.a.	10	1,200	
C9 DSSS / OFDM	16–30	14	100–1,000	n.a.	12 / 30	n.a.	6–60	up to 10,000	

Remarks: a: data from [63], b: OEM version w/o housing available, c: values partially refer to bit rate (w/o encoding, etc.), d: under development, e: without transducer.

## ACKNOWLEDGMENTS

The authors would like to thank Lucas Bublitz for assistance with building modem prototypes and Timo Kortbrae for help with software implementation. Our sincere gratitude goes to TuS Finkenwerder for granting access to their jetties and facilities. We thank Sea&Sun Technologies for providing a CTD-48, and we thank the Institute of Mechanics and Ocean Engineering for supplying a HippoCampus and the Institute of Aircraft Production Technology for providing laser measurement devices. A very hearty thank you goes to Prof. Andree from TUHH for providing the high-precision GNSS unit and sacrificing a weekend for fixing an issue with remote server access for our experiments. Finally, we express our gratitude to Prof. Maehle and his group from the University of Lübeck for their generous support; in particular, we honor the support of Alexander Golkowski and Uli Behrje, and we thank Benjamin Meyer for providing all MONSUN-related artwork.

## REFERENCES

- [1] Muhammad Hamad Alizai, Olaf Landsiedel, Jö Agila Bitsch Link, Stefan Götz, and Klaus Wehrle. 2009. Bursty traffic over bursty links. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys'09)*. ACM, Berkeley, CA, 71–84.
- [2] Analog Devices. 2006. AD5245 Datasheet. Retrieved from <http://www.analog.com/media/en/technical-documentation/data-sheets/AD5245.pdf>.
- [3] Analog Devices. 2014. AD8031 Datasheet. Retrieved from [http://www.analog.com/media/en/technical-documentation/data-sheets/AD8031\\_8032.pdf](http://www.analog.com/media/en/technical-documentation/data-sheets/AD8031_8032.pdf).
- [4] Analog Devices. 2015. AD56x1 12-Bit nanoDAC with SPI Interface. Retrieved from [http://www.analog.com/media/en/technical-documentation/data-sheets/AD5601\\_5611\\_5621.pdf](http://www.analog.com/media/en/technical-documentation/data-sheets/AD5601_5611_5621.pdf).
- [5] Analog Devices. 2015. ADA4807 Datasheet. Retrieved from [http://www.analog.com/media/en/technical-documentation/data-sheets/ADA4807-1\\_4807-2\\_4807-4.pdf](http://www.analog.com/media/en/technical-documentation/data-sheets/ADA4807-1_4807-2_4807-4.pdf).
- [6] Analog Devices. 2015. ADG5401 High Voltage Latch-Up Proof, Single SPST Switch. Retrieved from <http://www.analog.com/media/en/technical-documentation/data-sheets/ADG5401.pdf>.
- [7] Aquarian Audio & Scientific. 2019. Aquarian AS-1 Hydrophone. Retrieved from <http://www.aquarianaudio.com/as-1-hydrophone.html>.
- [8] Guillermo Barrenetxea, François Ingelrest, Gunnar Schaefer, and Martin Vetterli. 2008. The hitchhiker’s guide to successful wireless sensor network deployments. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (SenSys’08)*. ACM, 43–56. DOI : <https://doi.org/10.1145/1460412.1460418>
- [9] Ulrich Behrje, Cedric Isokeit, Benjamin Meyer, and Erik Maehle. 2018. A robust acoustic-based communication principle for the navigation of an underwater robot swarm. In *Proceedings of the MTS/IEEE OCEANS Conference & Exposition*. IEEE, 5. DOI : <https://doi.org/10.1109/OCEANSKOBE.2018.8558871>
- [10] Bridget Benson, Y. Li, Ryan Kastner, B. Faunce, K. Domond, Donald Kimball, and C. Schurges. 2010. Design of a low-cost, underwater acoustic modem for short-range sensor networks. In *Proceedings of the MTS/IEEE Oceans Conference (OCEANS’10)*. IEEE, 9.
- [11] Blueprint SeaTrac. 2019. X110 Modem Transponder Beacon. Retrieved from <https://www.blueprintsubsea.com/pages/product.php?PN=BP00843>.
- [12] Carlo Alberto Boano, Simon Duquennoy, Anna Förster, Omprakash Gnawali, Romain Jacob, Hyung-Sin Kim, Olaf Landsiedel, Ramona Marfievici, Luca Mottola, Gian Pietro Picco, Xavier Vilajosana, Thomas Watteyne, and Marco Zimmerling. 2018. IoTBench: Towards a benchmark for low-power wireless networking. In *Proceedings of the IEEE Workshop on Benchmarking Cyber-Physical Networks and Systems (CPSBench’18)*. IEEE, 36–41. DOI : <https://doi.org/10.1109/CPSBench.2018.00013>
- [13] Pierre-Jean Bouvet and Alain Loussert. 2010. Capacity analysis of underwater acoustic MIMO communications. In *Proceedings of the IEEE OCEANS Conference & Exposition*. IEEE, 1–8. DOI : <https://doi.org/10.1109/OCEANSSYD.2010.5603661>
- [14] Leonid Brekhovskikh and Yury Lysanov. 1982. *Fundamentals of Ocean Acoustics*. Springer, New York, NY.
- [15] Michael S. Caley and Alec J. Duncan. 2016. Wide-band shallow acoustic channel simulation with realistic doppler and delay spreading for 3D evolving rough surfaces. In *Proceedings of the International Conference on Underwater Communications and Networking (UComms’16)*. IEEE, 5.
- [16] Gianni Cario, Alessandro Casavola, Marco Lupia, and Claudio Rosace. 2015. SeaModem: A low-cost underwater acoustic modem for shallow water communication. In *Proceedings of the MTS/IEEE Oceans Conference*. IEEE, 6.

- [17] Emrecan Demirors, Jiacheng Shi, Anh Duong, Neil Dave, Raffaele Guida, Bernard Herrera, Flavius Pop, Guofeng Chen, Cristian Casella, Sayedamirhossein Tadayon, Matteo Rinaldi, Stefano Basagni, Milica Stojanovic, and Tommaso Melodia. 2018. The SEANet project: Toward a programmable internet of underwater things. In *Proceedings of the IEEE Underwater Communications Conference and Workshop (UComms'18)*. IEEE, 5.
- [18] Emrecan Demirors, George Sklivanitis, G. Enrico Santagati, Tommaso Melodia, and Stella N. Batalama. 2018. A high-rate software-defined underwater acoustic modem with real-time adaptation capabilities. *IEEE Access* 6 (2018), 18602–18615. DOI : <https://doi.org/10.1109/ACCESS.2018.2815026>
- [19] Developic Subsea Systems. 2014. Compact Hydro Acoustic Modem HAM.BASE. Retrieved from [http://www.developic.de/wp-content/uploads/2014/04/HAM.Node\\_HAM.Base-04-2014.pdf](http://www.developic.de/wp-content/uploads/2014/04/HAM.Node_HAM.Base-04-2014.pdf).
- [20] Vladimir Djapic, Wenjie Dong, Daniele Spaccini, Gianni Cario, Alessandro Casavola, Petrika Gjanci, Marco Lupia, and Chiara Petrioli. 2016. Cooperation of coordinated teams of autonomous underwater vehicles. In *Proceedings of the 9th IFAC Symposium on Intelligent Autonomous Vehicles*. Elsevier, 1–6.
- [21] Marek Doniec, Iulian Topor, Mandar Chitre, and Daniela Rus. 2013. *Autonomous, Localization-Free Underwater Data Muling Using Acoustic and Optical Communication*. Springer International Publishing, Heidelberg, 841–857. DOI : [https://doi.org/10.1007/978-3-319-00065-7\\_56](https://doi.org/10.1007/978-3-319-00065-7_56)
- [22] DSPComm. 2019. Aquacomm Gen2: Next Generation Underwater Wireless Modem. Retrieved from <https://dspcommgen2.com/aquacomm-gen2-next-generation-acoustic-modem/>.
- [23] Melike Erol-Kantarcı, Hussein T. Mouftah, and Sema Oktug. 2011. A survey of architectures and localization techniques for underwater acoustic sensor networks. *IEEE Commun. Surv. Tutor.* 13, 3 (Sept. 2011), 487–502.
- [24] Evologics GmbH. 2018. Underwater Acoustic Modems. Retrieved from <http://www.evologics.de/en/products/acoustics/>.
- [25] Emad Felemban, Faisal Karim Shaikh, Umair Qureshi, Adil Sheikh, and Saad Qaisar. 2015. Underwater sensor network applications: A comprehensive survey. *Int. J. Distrib. Sens. Netw.* 2015 (Nov. 2015), 1–14. DOI : <https://doi.org/10.1155/2015/896832>
- [26] Lee Freitag, Matthew Grund, Sandipa Singh, Jim Partan, Peter Koski, and Keenan Ball. 2005. The WHOI micro-modem: An acoustic communications and navigation system for multiple platforms. In *Proceedings of the MTS/IEEE Oceans Conference*. IEEE, 1086–1092.
- [27] Eric Gallimore, Jim Partan, Ian Vaughn, Sandipa Singh, Jon Shusta, and Lee Freitag. 2010. The WHOI micromodem-2: A scalable system for acoustic communications and networking. In *Proceedings of the MTS/IEEE Oceans Conference*. IEEE, 7.
- [28] WHOI Acoustic Communications Group. 2019. WHOI Micromodem 2.0. Retrieved from <http://acomms.whoi.edu/micro-modem/>.
- [29] Camila M. G. Gussen, Paulo S. R. Diniz, Marcello L. R. de Campos, Wallace Martins, Felipe M. Costa, and Jonathan N. Gois. 2016. A survey of underwater wireless communication technologies. *J. Commun. Inf. Syst.* 31, 1 (Oct. 2016), 14. DOI : <https://doi.org/10.14209/jcis.2016.22>
- [30] Axel Hackbarth, Edwin Kreuzer, and Eugen Solowjow. 2015. HippoCampus: A micro underwater vehicle for swarm applications. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'15)*. IEEE, 2258–2263.
- [31] Xiao Han, Jing-wei Yin, Bing Liu, and Long-xiang Guo. 2019. MIMO underwater acoustic communication in shallow water with ice cover. *China Ocean Eng.* 33 (Apr. 2019), 237–244. DOI : <https://doi.org/10.1007/s13344-019-0023-7>
- [32] John Heidemann, Milica Stojanovic, and Michele Zorzi. 2012. Underwater sensor networks: Applications, advances, and challenges. *Philos. Trans. Roy. Soc.-A* 370, 1958 (Jan. 2012), 158–175.
- [33] Jan Heitmann, Fabian Steinmetz, and Christian Renner. 2018. Self-localization of micro AUVs using a low-power, low-cost acoustic modem. In *Proceedings of the MTS/IEEE OCEANS Conference & Exposition (OCEANS'18)*. IEEE, 9.
- [34] Hydromea. 2019. Vertex AUV System Overview. Retrieved from <https://www.hydromea.com/vertex-autonomous-underwater-swarm/>.
- [35] Slamet Indriyanto and Ian Yosef Matheus Edward. 2018. Ultrasonic underwater acoustic modem using frequency shift keying (FSK) modulation. In *Proceedings of the 4th International Conference on Wireless and Telematics (ICWT'18)*. IEEE, 1–4.
- [36] Cedric Isokeit, Benjamin Meyer, and Erik Maehle. 2017. Cooperative swarm behaviour for in situ underwater environmental measurements. In *Proceedings of the MTS/IEEE Oceans Conference (OCEANS'17)*. IEEE, 6.
- [37] Weihua Jiang, Feng Tong, and Yuehai Zhou. 2016. R&D of a spread spectrum acoustic communication modem with ranging capability. In *Proceedings of the 11th ACM International Conference on Underwater Networks & Systems (WUWNet'16)*. ACM, 4.
- [38] Chien-Chi Kao, Yi-Shan Lin, Geng-De Wu, and Chun-Ju Huang. 2017. A comprehensive study on the internet of underwater things: Applications, challenges, and channel models. *MDPI Sensors* 17 (June 2017), 20, Issue 7. DOI : <https://doi.org/10.3390/s17071477>

- [39] Giannis Kazdaridis, Stratos Keranidis, Polychronis Symeonidis, Paulo Sousa Dias, Pedro Goncalves, Bruno Loureiro, Petrika Gjanci, and Chiara Petrioli. 2017. EVERUN: Enabling power consumption monitoring in underwater networking platforms. In *Proceedings of the 11th Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization (WiTECH'17)*. ACM, 8.
- [40] Konstantin Kebkal and Rudolf Bannasch. 2002. Sweep-spread carrier for underwater communication over acoustic channels with strong multipath propagation. *J. Acoust. Soc. Amer.* 112 (Dec. 2002), 2043–52. DOI : <https://doi.org/10.1121/1.1504855>
- [41] Konstantin Kebkal, Andrei Mashoshin, Sergey Yakovlev, Oleksiy Kebkal, and Veronika Kebkal. 2018. Phase estimation error of a PSK underwater acoustic signal in presence of multipath and volume scattering. In *Proceedings of the 4th Underwater Communications and Networking Conference (UComms'18)*. IEEE, 1–5. DOI : <https://doi.org/10.1109/UComms.2018.8493201>
- [42] Linghe Kong, Yifeng Cao, Liang He, Guihai Chen, Min-You Wu, and Tian He. 2019. Multi-rate selection in ZigBee. *IEEE/ACM Trans. Netw.* 27, 3 (June 2019), 1055–1068. DOI : <https://doi.org/10.1109/TNET.2019.2913014>
- [43] Edwin Kreuzer and Eugen Solowjow. 2018. Learning environmental fields with micro underwater vehicles: A path integral-Gaussian Markov random field approach. *Auton. Rob.* 42, 4 (Apr. 2018), 761–780. DOI : <https://doi.org/10.1007/s10514-017-9685-2>
- [44] Linear Technology. 2015. LTC3265 Low Noise Dual Supply with Boost and Inverting Charge Pumps. Retrieved from <http://www.linear.com/docs/46956>.
- [45] Jaime Lloret. 2013. Underwater sensor nodes and networks. *Sens. Spec. Iss. Underw. Sens. Nodes Underw. Sens. Netw.* 13, 9 (Sept. 2013), 11782–11796.
- [46] Maxim Integrated. 2014. MAX14756–MAX14758 Quad SPST +70V Analog Switches. Retrieved from <https://datasheets.maximintegrated.com/en/ds/MAX14756-MAX14758.pdf>.
- [47] Benjamin Meyer, Cedrik Isokeit, Erik Maeble, and Burkard Baschek. 2017. Using small swarm-capable AUVs for submesoscale eddy measurements in the Baltic Sea. In *Proceedings of the MTS/IEEE Oceans Conference (OCEANS'17)*. IEEE, 5.
- [48] Benjamin Meyer, Christian Renner, and Erik Maeble. 2016. Versatile sensor and communication expansion set for the autonomous underwater vehicle MONSUN. In *Proceedings of the 19th International Conference on Climbing and Walking Robots and Support Technologies for Mobile Machines (CLAWAR'16)*. World Scientific, 8.
- [49] Nils Morozs, Paul D. Mitchell, Yuriy Zakharov, Rahul Mourya, Yvan R. Petillot, Tyler Gibney, Mauro Dragone, Benjamin Sherlock, Jeffrey A. Neasham, Charalampos C. Tsimenidis, Mohammad E. Sayed, Alistair C. McConnell, Simona Aracri, and Adam A. Stokes. 2018. Robust TDA-MAC for practical underwater sensor network deployment: Lessons from USMART sea trials. In *Proceedings of the 13th ACM International Conference on Underwater Networks & Systems (WUWNet'18)*. ACM, 8.
- [50] Nusrat Nowsheen, Craig Benson, and Michael Frater. 2010. A high data-rate, software-defined underwater acoustic modem. In *Proceedings of the MTS/IEEE Oceans Conference (OCEANS'10)*. IEEE, 5.
- [51] Guillem Palou and Milica Stojanovic. 2009. Underwater acoustic MIMO OFDM: An experimental analysis. In *Proceedings of the MTS/IEEE OCEANS Conference & Exposition*. IEEE, 1–8. DOI : <https://doi.org/10.23919/OCEANS.2009.5422153>
- [52] Konstantinos Pelekanakis and Mandar Chitre. 2015. Robust equalization of mobile underwater acoustic channels. *IEEE J. Oceanic Eng.* 40, 4 (Oct. 2015), 775–784.
- [53] Chiara Petrioli. 2017. SUNRISE: Building the Internet of Underwater Things (FP7 Research Project). Retrieved from <http://fp7-sunrise.eu>.
- [54] David Pinto, Sadraque S. Viana, Jose Augusto M. Nacif, Luiz F. M. Vieira, Marcos A. M. Vieira, Alex B. Vieira, and Antonio O. Fernandes. 2012. HydroNode: A low cost, energy efficient, multi purpose node for underwater sensor networks. In *Proceedings of the 37th IEEE Conference on Local Computer Networks (LCN'12)*. IEEE, 148–151.
- [55] Madhuri Rao, Narendra Kumar Kamila, and Kulamala Vinod Kumar. 2016. Underwater wireless sensor network for tracking ships approaching harbor. In *Proceedings of the IEEE International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES'16)*. IEEE, 1872–1876. DOI : <https://doi.org/10.1109/SCOPES.2016.7955771>
- [56] Christian Renner. 2017. Packet-based ranging with a low-power, low-cost acoustic modem for micro AUVs. In *Proceedings of the 11th International ITG Conference on Systems, Communications and Coding (SCC'17)*. VDE, 6.
- [57] Christian Renner, Jan Heitmann, Erik Maeble, Cedric Isokeit, Benjamin Meyer, Philipp Rühmer, Uwe Leineweber, Peter Gutt, and Andrea Desel. 2017. *Automatische Ortung von Gefahrenstoffen im Hafenbecken (Automatic Detection of Hazardous Contaminants in the Port Basin)*. Technical Report. Hamburg University of Technology, Hamburg, Germany.
- [58] Clark Richards, Merle Pittman, Kirk Phelan, Shannon Nudds, and Jim Hamilton. 2017. The Barrow Strait real time observatory: Under-ice monitoring in the Canadian high Arctic. In *Proceedings of the International Conference on Underwater Networks & Systems (WUWNET'17)*. ACM, Article 8, 7 pages. DOI : <https://doi.org/10.1145/3148675.3152195>

- [59] Riptide Autonomous Solutions. 2019. Micro-UUV. Retrieved from <https://riptideas.com/micro-uuv/>.
- [60] Thomas Schmickl, Ronald Thenius, Christoph Möslinger, Serge Kernbach, Tobias Dipper, Donny Sutantyo, Jon Timmis, Andy Tyrrell, Mark Read, James Hilder, Cesare Stefanini, Luigi Manfredi, Stefano Orofino, Jose Halloy, and Alexandre Campo. 2011. CoCoRo—The self-aware underwater swarm. In *Proceedings of the 5th IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO'11)*. IEEE, 120–126.
- [61] Henrik Schmidt and Toby Schneider. 2016. Acoustic communication and navigation in the new Arctic: A model case for environmental adaptation. In *Proceedings of the International Conference Underwater Communications and Networking (UComms'16)*. IEEE, 4.
- [62] Sea & Sun Technology GmbH. 2019. CTD48 Multiparameter Online Probe. Retrieved from <https://www.sea-sun-tech.com/product/multiparameter-probe-ctd-48/>.
- [63] Sandra Sendra, Jaime Lloret, Jose Miguel Jimenez, and Lorena Parra. 2016. Underwater acoustic modems. *IEEE Sensors* 16, 11 (2016), 4063–4071.
- [64] Alberto Signori, Fabian Steinmetz, Filippo Campagnaro, Davide Zordan, Michele Zorzi, and Christian Renner. 2018. Poster: Underwater communications for the robotic vessels as-a-service project. In *Proceedings of the 13th ACM International Conference on Underwater Networks & Systems (WUWNet'18)*. ACM, 2.
- [65] Sonardyne. 2019. Underwater Acoustic Modem 6. Retrieved from <https://www.sonardyne.com/product/underwater-acoustic-modems/>.
- [66] Fabian Steinmetz, Jan Heitmann, and Christian Renner. 2018. A practical guide to chirp spread spectrum for acoustic underwater communication in shallow waters. In *Proceedings of the 13th ACM International Conference on Underwater Networks & Systems (WUWNet'18)*. ACM, 8.
- [67] Fabian Steinmetz and Christian Renner. 2019. Resilience against shipping noise and interference in low-power acoustic underwater communication. In *Proceedings of the MTS/IEEE OCEANS Conference & Exposition (OCEANS'19)*. IEEE, 10.
- [68] STMicroelectronics. 2019. STM32F446RE Cortex M4 (Rev. 7). Retrieved from <http://www.st.com/resource/en/datasheet/stm32f446re.pdf>.
- [69] Milica Stojanovic and James Preisig. 2009. Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Commun. Mag.* 47, 1 (Jan. 2009), 84–89.
- [70] Antonio Sánchez, Sara Blanc, Pedro Yuste, Angel Perles, and Juan José Serrano. 2011. A low cost and high efficient acoustic modem for underwater sensor networks. In *Proceedings of the MTS/IEEE Oceans Conference (OCEANS'11)*. IEEE, 10.
- [71] Qiuyang Tao, Yuehai Zhou, Feng Tong, Aijun Song, and Fumin Zhang. 2018. Evaluating acoustic communication performance of micro AUV in confined space. In *Proceedings of the MTS/IEEE OCEANS Conference & Exposition (OCEANS'18)*. IEEE, 6.
- [72] Teledyne Benthos. 2019. ATM-903 Series (OEM). Retrieved from <http://www.teledynemarine.com/903-series-atm-903/>.
- [73] Texas Instruments. 2007. TI TPS73150 Datasheet. Retrieved from <http://www.ti.com/lit/gpn/tps73150-ep>.
- [74] Texas Instruments. 2011. LMR64010 40V, 1A Step-Up Voltage Regulator. Retrieved from <http://www.ti.com/lit/gpn/lmr64010>.
- [75] Texas Instruments. 2014. LMR16006 60V 0.6A Buck Regulators With High Efficiency ECO Mode. Retrieved from <http://www.ti.com/lit/gpn/lmr16006>.
- [76] Texas Instruments. 2014. TI LM6134 Datasheet. Retrieved from <http://www.ti.com/lit/gpn/lm6134>.
- [77] Texas Instruments. 2014. TI LM6154 Datasheet. Retrieved from <http://www.ti.com/lit/gpn/lm6154>.
- [78] Texas Instruments. 2015. TI TPS62120 Datasheet. Retrieved from <http://www.ti.com/lit/gpn/tps62120>.
- [79] Texas Instruments. 2016. OPA55x High-Voltage, High-Current Operational Amplifiers. Retrieved from <http://www.ti.com/lit/gpn/opa551>.
- [80] Texas Instruments. 2016. TI ADC121S101 Datasheet. Retrieved from <http://www.ti.com/lit/gpn/adc121s101>.
- [81] Texas Instruments. 2016. TI OPA365 Datasheet. Retrieved from <http://www.ti.com/lit/gpn/opa365>.
- [82] Texas Instruments. 2017. OPAX172 36V, Single-Supply, 10MHz, Rail-to-Rail Output Operational Amplifiers. Retrieved from <http://www.ti.com/lit/ds/symlink/opa4172-q1.pdf>.
- [83] Texas Instruments. 2017. TPS54202 4.5-V to 28-V Input, 2-A Output, EMI Friendly Synchronous Step Down Converter. Retrieved from <http://www.ti.com/lit/ds/symlink/tps54202.pdf>.
- [84] Dag Tollefse and Stan E. Dosso. 2017. Source localization with multiple hydrophone arrays via matched-field processing. *IEEE J. Ocean. Eng.* 42, 3 (July 2017), 654–662. DOI: <https://doi.org/10.1109/JOE.2016.2615720>
- [85] Giovanni Toso, Paolo Casari, and Michele Zorzi. 2014. The effect of different attenuation models on the performance of routing in shallow-water networks. In *Proceedings of the International Conference Underwater Communications and Networking (UComms'14)*. IEEE, 5.

- [86] Tritech International Limited. 2019. Micron Modem—Underwater Acoustic Modem. Retrieved from <https://www.tritech.co.uk/product/micron-data-modem>.
- [87] Gurkan Tuna, Orhan Arkoc, and Kayhan Gulez. 2013. Continuous monitoring of water quality using portable and low-cost approaches. *Int. J. Distrib. Sens. Netw.* 9, 6 (2013), 249598. DOI : <https://doi.org/10.1155/2013/249598>
- [88] Robert J. Urick. 1983 (2013). *Principles of Underwater Sound 3rd Edition*. Peninsula Publishing, Westport, CT.
- [89] George S. K. Wong and Shi ming Zhu. 1995. Speed of sound in seawater as a function of salinity, temperature, and pressure. *J. Acoust. Soc. Amer.* 97 (1995), 1732–1736. Issue 3.
- [90] Muhammad Yousuf Irfan Zia, Pablo Otero, and Javier Poncela. 2018. Design of a low-cost modem for short-range underwater acoustic communications. *Wirel. Pers. Commun.* 101, 1 (1 July 2018), 375–390.

Received February 2019; revised December 2019; accepted December 2019