

Design of A Software-defined Underwater Acoustic Modem with Real-time Physical Layer Adaptation Capabilities

Emrecaan Demirors,* George Sklivanitis,† G. Enrico Santagati,* Tommaso Melodia,*
and Stella N. Batalama,†

*Department of Electrical and Computer Engineering, Northeastern University, Boston, MA
Email: {edemirors, santagati, melodia}@ece.neu.edu

†Department of Electrical Engineering, The State University of New York at Buffalo, Buffalo, NY
Email: {gsklivan, batalama}@buffalo.edu

ABSTRACT

This article describes the design of a custom software-defined modem with adaptive physical layer for underwater acoustic (UWA) communications. The modem consists of a commercial software-defined radio (SDR) interfaced with a wide-band acoustic transducer through amplifying circuitry. With this custom-built platform, we focus on the unique physical layer challenges of the underwater acoustic channel to demonstrate the benefits of real-time adaptation in such rapidly varying environments. We first focus on an Orthogonal-Frequency-Division-Multiplexing (OFDM) transmission scheme. In particular, for the forward link, we consider and implement a high-data rate Zero-Padded OFDM (ZP-OFDM) physical layer with a superimposed convolutional error-correction coding scheme. ZP-OFDM offers high reconfigurability in terms of number of OFDM subcarriers, modulation type (e.g., BPSK, QPSK), and error-correction coding rate. Real-time adaptation at the transmitter is achieved through a robust feedback link based on a binary chirp spread-spectrum modulation (B-CSS). We demonstrate that joint real-time adaptation of system parameters such as modulation constellation and channel coding rate leads to significant data rate increase under preset bit-error-rate (BER) constraints. Moreover, in the same context, we present for the first time a seamless switch of our SDR transmitter between different signaling technologies such as OFDM and direct-sequence spread-spectrum (DS-SS).

Keywords

Underwater modem, underwater networks, software-defined radio, software-defined networks.

Acknowledgement: This work is based upon work supported in part by the US National Science Foundation under grants CNS-1422874 and CNS-1126357.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
WUWNET '14, November 12 - 14 2014, Rome, Italy
Copyright 2014 ACM 978-1-4503-3277-4/14/11 ...\$15.00
<http://dx.doi.org/10.1145/2671490.2674473>

1. INTRODUCTION

Underwater acoustic (UWA) communications in combination with software-defined radio (SDR) techniques have lately attracted considerable attention, mainly because of the flexibility that SDRs may offer in rapidly time varying environments. Typical challenges in underwater communications include high path loss, severe multipath, Doppler spread, limited bandwidth, and fast time-varying channel [1]. On the other hand, SDR techniques have been envisioned as a powerful tool for designing cognitive, intelligently adaptive radio links in radio frequency (RF) communications [2–4], enabling this way efficient spectrum usage. The work in [5, 6] outlines potential benefits of using SDRs in UWA communication, and discusses how “cognitive” and adaptive signal processing techniques may be useful paradigms to shift away from hardware-based modems.

Because of the severe multipath channel variations and the spatially and temporally variable bandwidth [1] in the UWA, an adaptive software controlled, easily reconfigurable transceiver is highly desirable. Commercially available acoustic modems offer limited customization capabilities, while adding functionalities to proprietary systems is usually difficult and expensive. Conversely, working with a platform where most of the functionalities are controlled in software allows researchers to freely investigate designs where different parameters can be reconfigured at runtime (e.g., modulation, carrier frequency, coding). Prior work [7–9] on adaptive PHY schemes considers UWA modems with limited number of operating modes that offer data recording capabilities while they consider adaptation to environmental variations in the form of extensive offline simulations. The works in [10, 11] discuss in detail design considerations and present experimental results of a real-time adaptive Orthogonal-Frequency-Division-Multiplexing (OFDM) technology. However, the fixed hardware of these off-the-shelf acoustic modems supports only limited reconfigurations. Therefore, the need for a modem that supports full reconfigurability to enable cognitive, runtime-adaptive communication systems is evident [12, 13]. Moreover, such SDR-based modem would offer the ability to incorporate further algorithmic developments to improve its performance through software updates.

Based on these considerations, we present our design of a custom, high-data-rate, highly reconfigurable software-defined underwater modem and demonstrate the performance benefits of reconfiguration, real-time adaptation, and physical layer agility in rapidly varying environments such as the UWA channel. The software-defined underwater modem consists of a Universal Software Radio Peripheral (USRP

N210) interfaced with a Teledyne RESON TC4013 miniature wideband hydrophone that is used both for projecting and receiving sound in a time division fashion. The software-defined functionalities are implemented mainly in the GNU Radio framework that runs on an external host-PC. For the forward link, we implement a high-data rate ZP-OFDM physical layer with a superimposed convolutional error-correction coding scheme. The modem offers higher or similar data rates compared to commercial and experimental acoustic modems [1] and other custom built modems [13] offering high reconfigurability in terms of the number of OFDM subcarriers, modulation type (e.g., BPSK, QPSK), and error-correction coding rate. To enable efficient real-time adaptation both at the transmitter and receiver, we also design a new robust feedback link based on binary chirp spread-spectrum modulation (B-CSS). Finally, we demonstrate seamless switch between different communication technologies to highlight the flexibility of the proposed modem in physical layer adaptation.

The custom built underwater modems are deployed and tested both in a water test tank and in a shallow outdoor lake. First, we evaluate the proposed system setup in terms of bit-error-rate (BER) as a function of signal-to-interference-plus-noise-ratio (SINR) at the receiver, for various modulation schemes and error-correction coding rates. Second, we demonstrate real-time data rate performance versus power consumption tradeoff points under predefined BER reliability constraints. Finally, we demonstrate for the first time a seamless interchange between ZP-OFDM and direct-sequence spread-spectrum (DS-SS) communication technologies at run-time.

To sum up, the major contributions of this paper are:

- Design of a high-data-rate, highly reconfigurable software-defined modem based on USRP;
- design of a new robust chirp-based feedback channel;
- design of mechanisms for flexible adaptation of PHY parameters (e.g., modulation, error-correction coding rate) and seamless switch between different communication technologies.

The rest of the paper is organized as follows. In Section 2, we describe the modem architecture and discuss the design and synthesis of a software-defined underwater modem from first principles. Section 3 analyzes the physical layer schemes for the forward and feedback communication links. Section 4 presents performance evaluation results from tests conducted in both an indoors located water test tank and outdoor experiments conducted in a lake. Finally, a few concluding remarks are provided in Section 5.

2. MODEM ARCHITECTURE

The custom SDR-based acoustic modems consist of (i) a USRP N210, (ii) a host Linux-PC, (iii) a power amplifier/voltage preamplifier, (iv) an electronic switch, and (v) an acoustic transducer. The proposed hardware architecture of the modem is illustrated in Fig. 1.

USRP N210. We selected USRP N210, which is a commercially available, FPGA-based, SDR platform that offers a wide range of radio front-ends, through daughterboards, covering the frequency spectrum from DC to 5 GHz. Based on our project requirements, we selected LFTX and LFRX daughterboards (DC – 30 MHz) that enable the development of a half-duplex transceiver operating in the frequency range of the selected acoustic transducer from 1 Hz

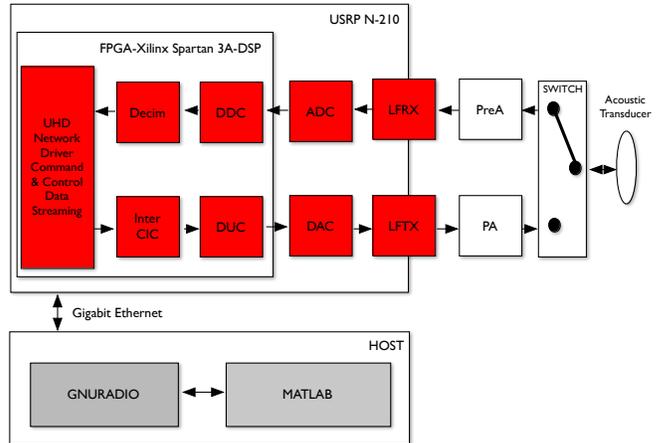


Figure 1: Hardware architecture of the software-defined acoustic modem (SDAM).

to 170 kHz. The USRP’s motherboard is equipped with an analog-to-digital converter (ADC) and a digital-to-analog-converter (DAC) that are both controlled by a 100 MHz master clock. The sample rate of incoming digital samples (from ADC) and outgoing samples (to DAC) is fixed at 100 Msample/s, while the FPGA digitally interpolates/decimates to match the hardware sample rate to the rate requested by the user. High rate baseband signal processing can be conducted either in the FPGA (Xilinx Spartan 3A-DSP3400) or in the host-PC, which is connected to the radio through a Gigabit Ethernet (GigE) connection.

To implement transmitter and receiver algorithms, we used a free and open-source software framework called GNU Radio that is commonly used (i) to drive the USRP operations from the host-PC, (ii) as well as implementing signal processing operations in combination with MATLAB scripting language. GNU Radio offers a broad set of signal processing blocks, implemented in C++ that can be used to develop a large variety of wireless communications applications. These basic C++ blocks are usually wrapped into Python classes to be instantiated from Python scripts, or used as building blocks of a communication flowgraph from a graphic user interface. We selected GNU Radio because of the already available communication building blocks that allow rapid prototyping of PHY layer schemes, and also because of the capability to easily create new custom blocks and thus create customized transceiver designs. Both forward and feedback links discussed in this paper are implemented using GNU Radio blocks. Some of the receiver functionalities combine both GNU Radio blocks and MATLAB scripting language.

Power Amplifier, Voltage Preamplifier, and Switch.

To enhance the communication range of our system, we selected a commercial linear wideband power amplifier (PA), Benthowave BII – 5002. The PA has maximum output power of 192 dB, can support up to 300 kHz bandwidth and is used for amplifying the output power of the LFTX daughterboard (i.e., 3mW). Half-duplex operation between TX / RX with a single acoustic transducer is achieved through a commercial electronic switch, Mini-Circuits ZX80 – DR230+. It provides low insertion loss and very high isolation over the frequency range of 0 – 3 GHz. The switch is controlled

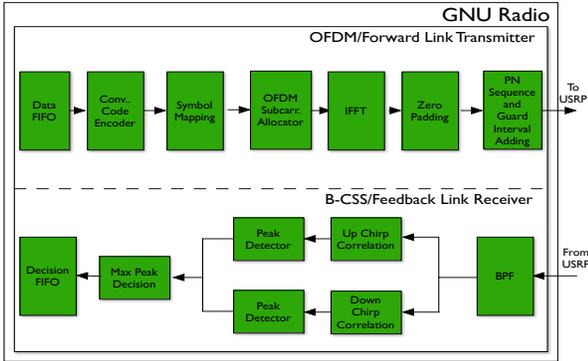


Figure 2: Block diagram of the ZP-OFDM transmitter and B-CSS receiver.

through the General Purpose Input/Output (GPIO) digital pins available on the LFTX and LFRX daughterboards. At the receiver side of the switch, we connected a voltage preamplifier (PreA), Teledyne RESON VP2000, that offers low-noise performance in the desired frequency range with a range of bandpass filters, and adjustable gain selection.

Acoustic Transducers. We used acoustic transducers (receiver hydrophones) manufactured by Teledyne RESON TC4013, which offer an operational frequency range from 1 Hz to 170 kHz. These transducers were selected mainly because of the relatively wide frequency bands that they can support, thus allowing high data rate communications and enabling the implementation of a variety of physical layer schemes. They provide receiving sensitivity of -211 [dB re $1\mu\text{Pa}$ at 1 m] that is flat over the operational frequency range and maximum transmitting sensitivity of 130 [dB re $1\mu\text{Pa}/\text{V}$ at 1 m] at 100 kHz. Moreover, the acoustic transducers have omnidirectional horizontal and 270° vertical directivity patterns.

3. PHYSICAL LAYER ADAPTATION

In this section, we describe the developed mechanisms for adaptation of PHY parameters and technologies that our modem can support in real-time, according to a predefined decision metric. A chirp-based feedback link is also defined to provide a reliable, low-data rate method for updating the transmitter parameters and switching seamlessly between different technologies upon the decision taken at the receiver.

OFDM PHY parameter adaptation. OFDM has been extensively used in UWA communications due to its robustness against frequency selective channels with long-delay spreads. Fig. 2 depicts a GNU Radio flowgraph that implements a K -subcarrier, ZP-OFDM scheme used in this setup. First, information bits are provided to the flowgraph through a data FIFO which is controlled by a Python thread. Bits are encoded by a convolutional encoder block and mapped into symbols according to the selected modulation scheme. Symbols are then allocated to subcarriers. Specifically, $K_P = K/4$ subcarriers are used as pilot subcarriers, K_D are used for data, and K_N are selected as null subcarriers. Afterwards, IFFT and zero-padding operations take place over blocks of incoming symbols from the subcarrier allocation block. Finally, a guard interval is added between each OFDM block and a PN sequence is transmitted at the beginning of each OFDM frame. Transmitter

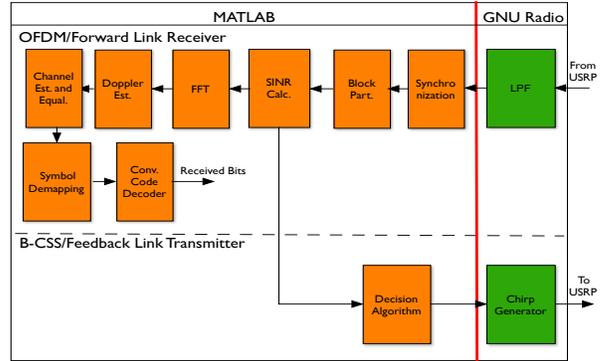


Figure 3: Block diagram of the ZP-OFDM receiver and B-CSS transmitter.

functionalities were implemented in GNU Radio by defining new and customizing already available communication building blocks.

On the receiver side (Fig. 3) a low-pass filter (LPF) is firstly used to filter out the out-of-band noise and interference. Incoming data are then pushed into a FIFO queue that is continuously polled by a MATLAB thread. The correlation properties of the PN sequence inserted at the transmitter are exploited for frame detection and receiver synchronization. Each OFDM frame is then partitioned into individual OFDM blocks and input SINR is estimated [11]. After FFT, we used the null subcarriers for Doppler scale estimation. The pilot symbols known at the transmitter are used for conducting channel estimation and symbol synchronization in the frequency domain [14]. Data symbols are then demapped and decoded through a convolutional code decoder and we acquire the received bits. The receiver architecture is based on a hybrid MATLAB/GNU Radio implementation.

In this setup, the adaptation mechanism is driven by the receiver node. According to the SINR estimated at the receiver, we select the optimal transmission strategy to satisfy a pre-defined BER threshold. This decision algorithm can be defined as a data rate R maximization problem

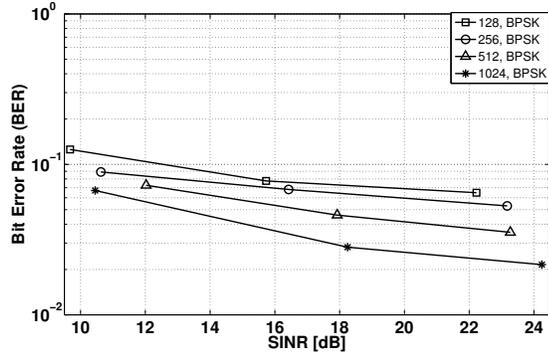
$$\begin{aligned} & \text{maximize} && R(M, C) \\ & \text{subject to} && BER_{th} \geq BER(SINR, M, C). \end{aligned}$$

where M is the modulation order, C is the error-correction coding rate, BER is a function of M , C and the $SINR$ estimated at the receiver, and BER_{th} is a pre-defined BER reliability threshold. Data rate R is defined as

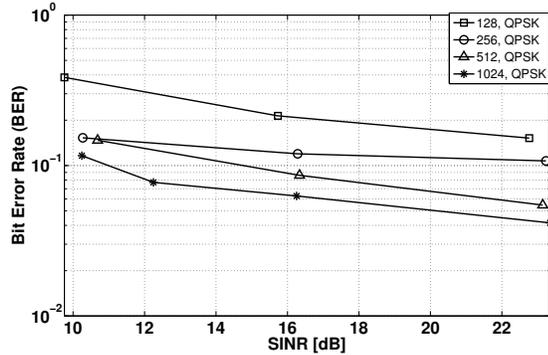
$$R(M, C) = \frac{K_D C \log_2(M)}{T_s + T_g},$$

where K_D is the number of data subcarriers, T_s is the symbol duration, and T_g is the duration of the guard interval. In Section 4, we will discuss how to select parameters M and C , for fixed K_D , T_s and T_g to maximize data rate under predefined BER reliability constraints.

This maximization problem is solved in a decision algorithm block which runs in MATLAB. The decisions are then sent to the GNU Radio flowgraph that implements the feedback transmission link (Fig. 3). As a result, the optimal strategy decided by the receiver, is communicated to the



(a) BER versus SINR for BPSK modulation with $K = 128, 256, 512,$ and 1024 .



(b) BER versus SINR for QPSK modulation with $K = 128, 256, 512,$ and 1024 .

Figure 4: BER versus SINR for different number of subcarriers and modulation schemes.

transmitter node through a reliable chirp-based feedback link. After successfully decoding the feedback message, the transmitter synchronizes with the decision taken at the receiver and adopts the same optimal strategy for its next data packet transmission. Parameter adaptation at the transmitter side is performed via a Python thread that controls the convolutional encoder, the symbol mapping, and the sub-carrier allocation blocks of the transmitter GNU Radio flow graph.

Adaptation between communication technologies.

To demonstrate switching between different communication technologies, we have additionally implemented DS-SS transmitter/receiver flowgraphs by developing custom communication blocks in the GNU Radio environment. Each transmission frame consists of P unmodulated pilot symbols $\in \{1\}$ and N data bits that are mapped into $\{+1, -1\}$ BPSK modulated symbols. Each symbol is then spread in L chips through a custom built block that outputs $(N + P)L$ chips in $\{+1, -1\}$. A RAKE receiver is also implemented in GNU Radio and pilot symbols are used for frame detection, channel estimation, and symbol synchronization. The decision mechanism adopted for this adaptation scenario is based on successful decoding of an incoming packet. More specifically, both OFDM and DS-SS flowgraph instances are defined in the software-defined modems and their operation is controlled by a software implemented multiplexer. The enabling signal for this multiplexer is controlled by a boolean variable

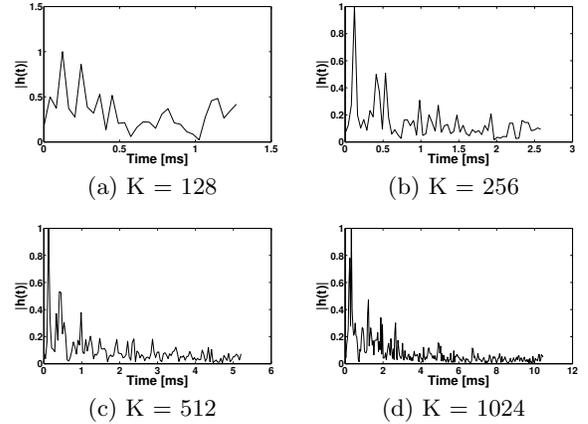


Figure 5: Channel estimation for different numbers of subcarriers.

that enables DS-SS transmission/reception upon successful reception of an OFDM packet and vice versa. The feedback link in this case is used as the carrier of the signal that will enable the seamless switch between the two communication technologies at the transmitter. The ability of the modems to support hybrid communication technologies may prove of significant importance in future deployments of “cognitive” underwater networks.

Chirp-Based Feedback Link. A reliable feedback link is crucial to support real-time adaptation. Potential failures in the feedback link lead to failures in the forward link as well. In fact, if an acknowledged packet containing feedback information is lost, the subsequent transmission on the forward link is also likely to fail because of inconsistency between the transmitter and the receiver physical layer configuration. To deal with this problem, we propose and implement a robust feedback link based on a binary chirp spread-spectrum (B-CSS) modulation, which is known to be resilient against the severe multipath and Doppler effects that characterize the UWA channel. Chirp signals have been presented in the UWA communications literature as highly reliable but low data rate alternatives [15]; they match well the requirements of feedback links, which need to be reliable but do not usually require high data rates. Moreover, the B-CSS signaling scheme offers a low complexity correlation-based receiver architecture with a simple detection scheme. To the best of our knowledge, we are the first to propose and test the use of chirp signals for feedback links in UWA communications.

A chirp signal is characterized by a time-varying instantaneous frequency, which changes in time from an initial value f_0 to a final value f_1 . In the time domain, the signal can be expressed as

$$c(t) = \begin{cases} A \cos(2\pi f_0 t + \pi \mu t^2) & 0 \leq t \leq T, \\ 0 & \text{otherwise,} \end{cases}$$

where A is the amplitude of the chirp, f_0 is the initial chirp frequency, $\mu = \frac{f_1 - f_0}{T}$ is the chirp frequency-variation rate, while T represents the chirp period. We refer to a chirp with parameter $\mu > 0$ as an up-chirp; otherwise, we call it a down-chirp. Up and down chirp signals are almost orthogonal to each other. The total bandwidth of the chirp signal can be obtained as $B = f_1 - f_0$.

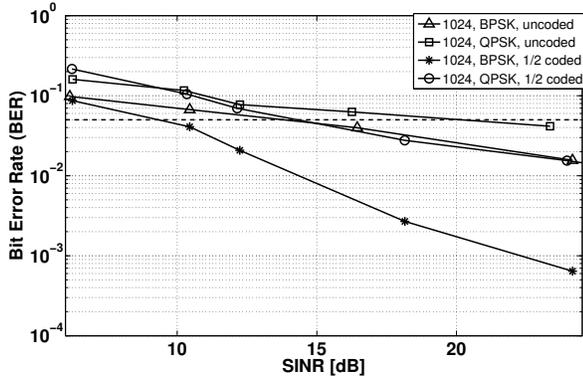


Figure 6: BER versus SINR for $K=1024$ with different modulation and coding.

In the feedback link, we leverage the quasi-orthogonality of up and down-chirps by encoding a ‘1’ bit with an up-chirp and a ‘0’ bit with a down-chirp. At the receiver, two parallel correlation filters with an up and down-chirp respectively, are used for decoding the incoming packets. For each feedback bit period, one of the two correlation filters will output a higher correlation peak, revealing the bit that was transmitted (‘1’ or ‘0’). Both transmitter and receiver functionalities of the feedback link are implemented in GNU Radio with custom C++ blocks. Block diagrams of the feedback receiver and transmitter are illustrated in Fig. 2 and 3 respectively.

4. PERFORMANCE EVALUATION

In this section, we demonstrate real-time implementation of OFDM physical layer parameter adaptation, and switch between different physical layer technologies, with both water tank and lake experiments.

Tank Tests. We conduct experiments in a water test tank of dimensions 8 ft \times 2.5 ft \times 2 ft. First, we evaluate the BER physical layer performance of the implemented ZP-OFDM scheme with respect to input SINR by varying i) number of subcarriers, ii) modulation scheme, and iii) error-correction coding rate. For this purpose, we use OFDM signals that occupy a bandwidth of $B = 24$ kHz at a carrier frequency $f_c = 100$ kHz. We selected the operating frequency band that results in the highest transmit power of our modem based on the characteristics of the transducer and amplifiers. We define a guard time $T_g = 15$ ms for each OFDM block and use $K = 128, 256, 512,$ and 1024 subcarriers with either BPSK or QPSK modulation and rate 1/2 convolutional error-correction codes.

Figure 4 shows BER performance versus SINR for different number of subcarriers and different modulation schemes (i.e., BPSK, QPSK). We observe that setups with higher number of subcarriers have better BER results. Increasing the number of subcarriers can offer better channel and Doppler scale estimates as the system can benefit from more pilot and null subcarriers. In Fig. 5, we observe that as the number of subcarriers increase, the channel estimation process exhibits higher resolution. On the other hand, increasing the number of subcarriers K , increases the symbol duration T_s which makes each OFDM block more vulnerable to channel variations. For example, in tank experiments, the

channel has an excessive number of channel taps (in the order of 300) and is characterized by few time variations. This condition makes a higher number of subcarriers a favorable solution.

Adaptive Coding and Modulation. By fixing the number of subcarriers to $K = 1024$ for better channel and Doppler estimates, we implemented an adaptive coding and modulation scheme to showcase the real-time PHY adaptation capabilities of the proposed system. For this purpose, we first conducted a series of experiments for different modulation schemes (i.e., $M = 2$ for BPSK and $M = 4$ for QPSK) and error-correction coding rates (i.e., no coding, $C = 1/2$). Figure 6 illustrates the BER performance versus SINR for four combinations of different modulations with coding schemes. As expected, at low SINR, lower-data rate modulation schemes, e.g., BPSK, have better BER performance as compared to higher-data rate modulation schemes, e.g., QPSK. The dashed line in Fig. 6 indicates a BER threshold at 5×10^{-2} , which is selected to be the minimum BER requirement for this set of experiments. Thus, according to the SINR estimated at the receiver, we adaptively select M and C such that this BER requirement is satisfied and the data rate is maximized.

We verify the effectiveness of real-time adaptation by comparing data rate and BER results of an adaptive scheme with a fixed scheme as a function of the estimated SINR. Fig. 7 shows experimental results for these two experiments. Each experiment includes transmission of 16 OFDM packets. The first 8 packets are transmitted with relatively low power compared to the last 8. Figure 7 illustrates the effect of varying the power profile of packets.

In particular, in both experiments, software-defined underwater acoustic modems started their transmission with the highest possible data rate (i.e., uncoded QPSK modulation). In the second packet transmission, the adaptation mechanism updates the modulation type and error-correction code and chooses a lower data rate scheme to fulfill the BER threshold requirements (5×10^{-2}). The adaptation decision is taken at the receiver node and communicated to the transmitter through the chirp-based feedback link. The fixed scheme keeps using the same high data rate scheme, which results in a BER higher than the predefined threshold. The SINR profiles remain constant between the 2nd and 8th packet transmissions, which proves that the adaptive scheme keeps satisfying the BER constraints, in contrast with the fixed scheme. Transmission power increases for the 9th packet transmission, as it can be also observed from the corresponding SINR estimation (Fig. 7). SINR estimates are thus higher for the 9th packet transmission, and the adaptive scheme chooses a higher data rate scheme for the next (i.e., 10th) packet transmission that still satisfies the BER reliability constraints.

In a different hypothetical scenario where transmission power was fixed, receiver SINR could vary due to multiuser interference or heavy impulsive noise. The proposed system would adapt either its modulation scheme or error-correction coding rate or both to satisfy BER performance constraints.

Lake Tests. A series of experiments was also conducted in Lake La Salle at the University at Buffalo. Testbed equipment such as power generators, power amplifiers, voltage preamplifiers, electronic switches, and USRP N210s (Fig. 8(a)) were deployed in Baird Point (Fig. 8(c)). RF cables of length 500ft were used to connect the software-defined radios with the underwater transducers deployed in the locations depicted in Fig. 8(b)-(c). The depth of the water in Lake La

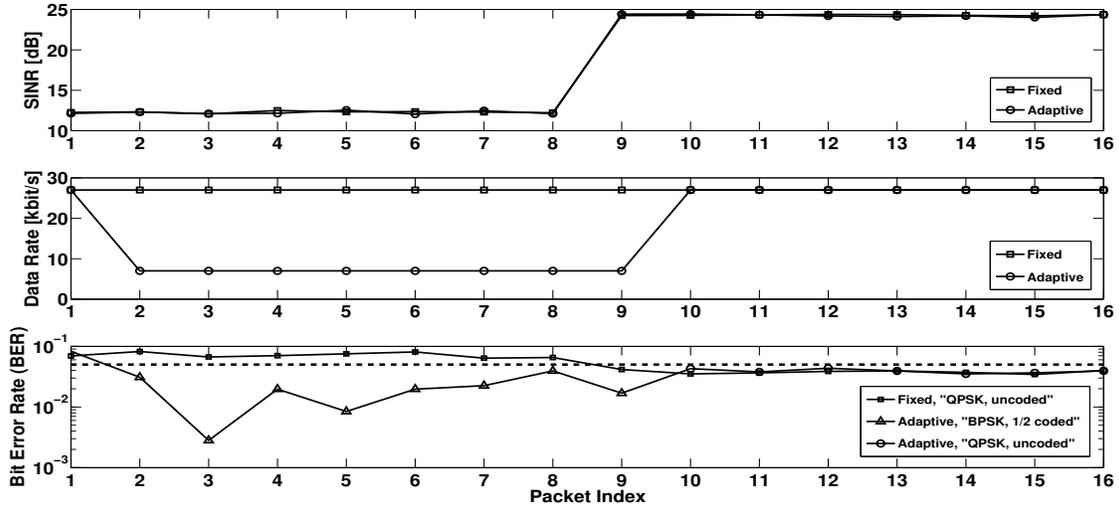
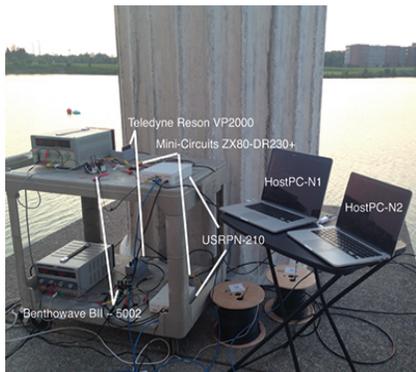


Figure 7: Comparison of Adaptive Scheme with Fixed Scheme in terms of SINR, data rate, and BER.



(a) Testbed setup located at Baird Point (Lake La Salle) using 500 ft RF cables to control each underwater transducer.



(b) Underwater transducers are held by the red buoys and deployed 322ft apart.



(c) Experimental setup in Google maps.

Figure 8: Experimental setup in Lake La Salle.

Salle is approximately 7 ft. Underwater transducers swung on their cable under the red buoys, which were anchored to the bottom. In these outdoor experiments, we used the same configuration as indoor test tank, i.e., OFDM signals of $B = 24$ kHz at carrier frequency $f_c = 100$ kHz, $K = 1024$ subcarriers, BPSK or QPSK modulation, and rate 1/2 convolutional error-correction codes.

Figure 10 presents experimental BER results versus SINR for a fixed number of $K = 1024$ OFDM subcarriers. Two different modulation schemes, BPSK and QPSK, either uncoded or coded with a 1/2 rate convolutional error-correction code, were tested in the shallow waters of the lake, between nodes N1-N2, which are positioned 322ft apart. We observe that BER performance results are significantly improved (especially for high SINR) compared to the test tank results in Fig. 6, mainly due to less severe multipath. Figure 11 illustrates 10 different realizations of channel estimates from independent experiments. We observe that only one distinct

and strong path is present. However, there is still a time-varying multipath effect that cannot be neglected because of the very shallow depth of the lake.

Adaptive Coding and Modulation. The real-time adaptation capabilities of our modem are tested in the lake environment by comparing the data rate and BER results of an adaptive modulation/coding scheme with a fixed modulation/coding scheme as a function of the estimated SINR. For this set of experiments we set the BER threshold to 10^{-3} . Figure 9 depicts experimental results from 21 OFDM packet transmissions with varying SINR profiles between 10 and 20 dB.

Both types of experiments, fixed and adaptive, start with the combination of modulation and coding schemes that result in maximum data rate, i.e., QPSK uncoded. For the fixed scheme, the data rate is maintained at its maximum value for all 21 packets. However, BER constraints are violated from the 4th up to the 17th packet. Specifically, BER

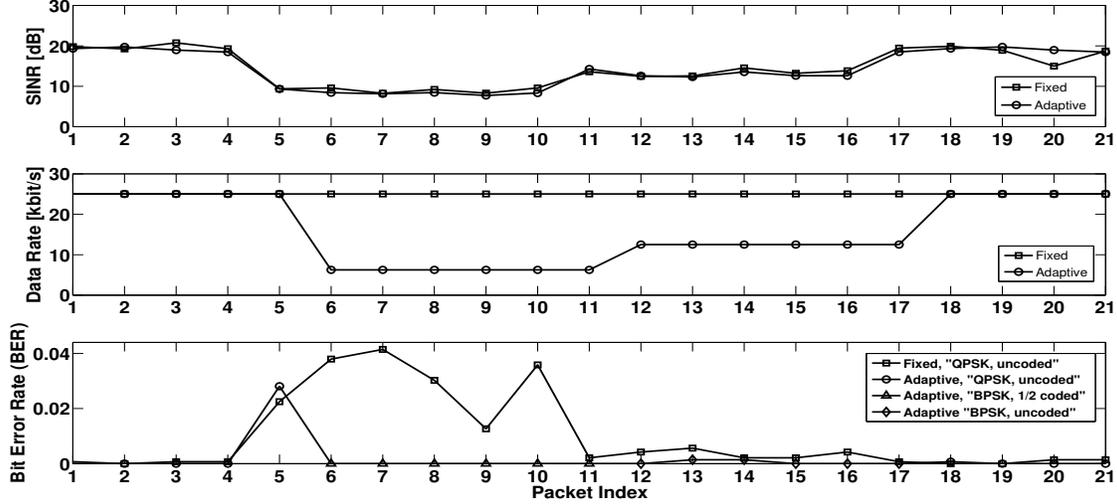


Figure 9: Comparison of Adaptive Scheme with Fixed Scheme in terms of SINR, data rate, and BER.

requirement is violated when the packets with SINR lower than 20 dB are received. Instead, in the adaptive modulation/coding mode, data rate is adjusted during low SINR profiles (i.e., <20 dB) to fulfill the preset BER constraints. In particular, modulation is changed from QPSK to BPSK as soon as BER increases to $\sim 2 \cdot 10^{-3}$ and 1/2 rate error-correction code is used up to the 11th packet transmission. SINR profile changes into higher values from packet 11 to 17. Therefore, the mode changes into BPSK with no error-correction coding to compensate for the data rate loss. After packet 17, SINR values return to the initial level of 20 dB, and the QPSK uncoded modulation/coding scheme is chosen.

Chirp-based feedback. Figure 12 depicts one of the received feedback packets that consists of two down and one up chirps. A bank of up/down chirp correlation filters at the receiver (Fig. 2) and peak detectors are used for decod-

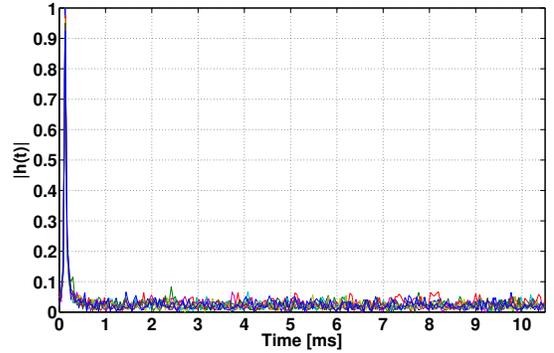


Figure 11: Channel estimates from 10 different packets during lake experiments.

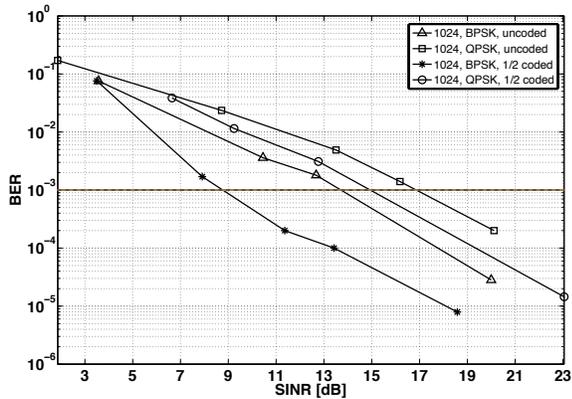


Figure 10: Lake experimental BER versus SINR for $K=1024$ with different modulation and coding.

ing the received bits. The decision block in Fig. 2 outputs '0' if correlation with a down chirp results in a higher peak, otherwise it outputs '1'. The outputs of both correlation filters are also depicted in Fig. 12. As a result, the decoded feedback packet in Fig. 12 corresponds to transmitted bits '010'. The format of the feedback packet used in our experiments assigns the first bit to switch between the two available communication technologies (OFDM, DS-SS), while the other two bits are used to select between BPSK, QPSK and coded or uncoded modulation in OFDM transmissions.

Switch between OFDM and DS-SS. Switching between DS-SS and OFDM technologies is illustrated in Fig. 13. During our lake experiments we were able to seamlessly switch between transmissions of OFDM packets (BPSK modulation, no error-correction code, $K = 1024$ subcarriers) and spread-spectrum packets (BPSK modulation, no error-correction code, and chip length $L = 31$), both of bandwidth $B = 24$ kHz at carrier frequency $f_c = 100$ kHz.

Our software modem was customized to initially transmit OFDM packets. Upon successful reception of an OFDM/DS-

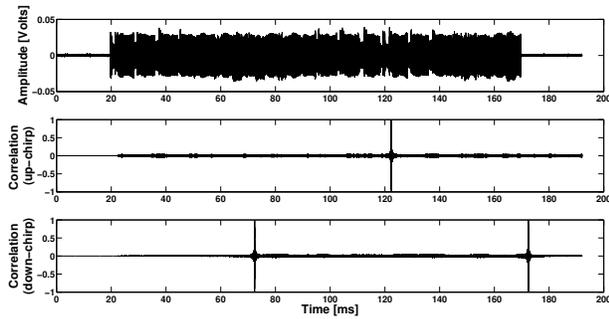


Figure 12: B-CSS feedback representing information bits ‘010’ and the respective correlations with up and down chirps.

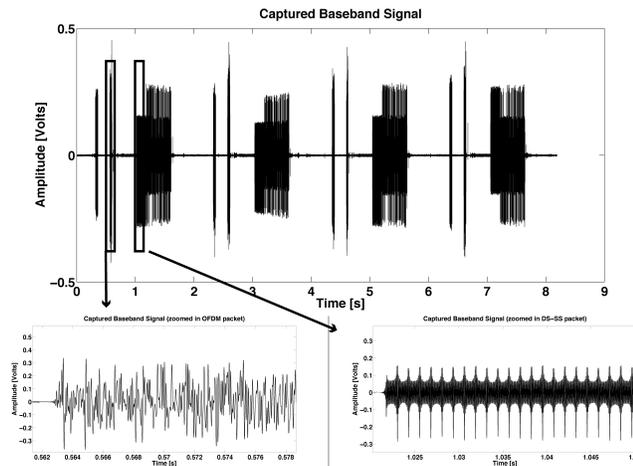


Figure 13: Received baseband signal consisting of OFDM and DS-SS packets.

SS packet, feedback for switching to DS-SS/OFDM packets is sent by the receiver. We select two different instances of a 8 s time window and focus on the received baseband signal. The bottom right part of Fig. 13 zooms on the received preamble part of the DS-SS packet (consisting of unmodulated bits); which clearly reveals the periodic pattern of the length- L spreading code. The bottom left part of Fig. 13 shows instead the OFDM packet.

5. CONCLUSIONS

We designed and built a high-data-rate, highly reconfigurable, software-defined underwater modem. Our modem was experimentally tested both in a water test tank and a lake and reached the data rate of ~ 28 kbit/s at a distance of 322 ft. We demonstrated the modems flexibility through i) PHY parameter adaptation in an OFDM technology and ii) seamless switch between OFDM and DS-SS communication technologies. Finally, we designed and implemented a robust chirp-based feedback channel that enables successful adaptation underwater.

6. REFERENCES

[1] T. Melodia, H. Kulhandjian, L. Kuo, and E. Demirors. Advances in Underwater Acoustic Networking. In

S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, editors, *Mobile Ad Hoc Networking: Cutting Edge Directions*, pages 804–852. John Wiley and Sons, Inc., Hoboken, NJ, second edition edition, 2013.

[2] G. Sklivanitis, E. Demirors, S. N. Batalama, D. A. Pados, T. Melodia, and J. D. Matyjas. Demonstration of all-spectrum cognitive channelization on GNU Radio and USRPN-210. In *Proc. of the NATO Symp. on “Cognitive Radio and Future Networks” (IST-123)*, The Netherlands, May 2014.

[3] L. Ding, K. Gao, T. Melodia, S. N. Batalama, D. A. Pados, and J. D. Matyjas. All-spectrum cognitive networking through joint distributed channelization and routing. *IEEE Transactions on Wireless Communications*, 12(11):5394–5405, November 2013.

[4] G. Kanke, S. N. Batalama, D. A. Pados, and J. D. Matyjas. Cognitive code-division channelization. *IEEE Transactions on Wireless Communications*, 10(4):1090–1097, April 2011.

[5] E. Jones. The application of software radio techniques to underwater acoustic communications. In *Proc. of IEEE OCEANS*, June 2007.

[6] E. M. Sözer and M. Stojanovic. Reconfigurable acoustic modem for underwater sensor networks. In *Proc. of ACM Intl. Workshop on Underwater Networks (WUWNet)*, Los Angeles, 2006.

[7] X. Huang and V. Lawrence. Bandwidth-efficient bit and power loading for underwater acoustic OFDM communication system with limited feedback. In *Proc. of Vehicular Technology Conf.*, Hungary, 2011.

[8] P. Qarabaqi and M. Stojanovic. Adaptive power control for underwater acoustic communications. In *Proc. of IEEE OCEANS*, Spain, June 2011.

[9] A. Radosevic, T. Duman, J. Proakis, and M. Stojanovic. Adaptive OFDM for underwater acoustic channels with limited feedback. In *Proc. of Signals, Systems and Computers (ASILOMAR)*, 2011.

[10] A. Radosevic, R. Ahmed, T. Duman, J. Proakis, and M. Stojanovic. Adaptive OFDM modulation for underwater acoustic communications: Design considerations and experimental results. *IEEE Journal of Oceanic Engineering*, 39(2):357–370, April 2014.

[11] L. Wan, H. Zhou, X. Xu, Y. Huang, S. Zhou, Z. Shi, and J.-H. Cui. Field tests of adaptive modulation and coding for underwater acoustic ofdm. In *Proc. of ACM Intl. Conf. on Underwater Networks and Systems (WUWNet)*, Taiwan, 2013.

[12] G. E. Santagati and T. Melodia. Sonar Inside Your Body: Prototyping Ultrasonic Intra-body Sensor Networks. In *Proc. of IEEE Conf. on Computer Communications (INFOCOM)*, Toronto, Canada, April 2014.

[13] D. Torres, J. Friedman, T. Schmid, and M. B. Srivastava. Software-defined underwater acoustic networking platform. In *Proc. of the ACM Intl. Workshop on UnderWater Networks (WUWNET)*, Berkeley, 2009.

[14] S. Zhou and Z.-H. Wang. *OFDM for Underwater Acoustic Communications*. John Wiley and Sons, Inc., 2014.

[15] L. LeBlanc, M. Singer, P.-P. Beaujean, C. Boubli, and J. Alleyne. Improved chirp FSK modem for high reliability communications in shallow water. In *Proc. of MTS/IEEE OCEANS*, 2000.