Real-time Through-water Video Streaming Using a High-rate Underwater Acoustic OFDM System

Peng Chen
School EECMS
Curtin University
Bentley, WA, Australia
peng.chen@curtin.edu.au

Yue Rong
School EECMS
Curtin University
Bentley, WA, Australia
y.rong@curtin.edu.au

Alec Duncan
CMST
Curtin University
Bentley, WA, Australia
a.j.duncan@curtin.edu.au

Sven Nordholm
School EECMS
Curtin University
Bentley, WA, Australia
s.nordholm@curtin.edu.au

Abstract—**Real-time video streaming through the underwater acoustic (UA) channel is challenging due to the limited bandwidth. In this paper, we present a high-rate, reconfigurable software-defined UA communication system that we recently developed, which is capable of real-time through-water video streaming. The transmitter consists of a universal software radio peripheral interfaced with a high-frequency transducer through a broadband impedance matching network designed in house. The transmitter and receiver signal processing algorithms are implemented using Python and run on external host computers. The system can reach a data rate of 445 kbps using a single transducer. The prototype system is tested in a UA communication experiment conducted in a hydroacoustic tank. Experimental results show that together with our video processing algorithms, this system can transmit real-time video with a high quality.**

Index Terms—**Software-defined radio, underwater acoustic communication, underwater video streaming.**

I. Introduction

The underwater acoustic (UA) channel is one of the most challenging media for wireless communication [1]. Due to its limited bandwidth, transmission of real-time video information through the UA channel is considered to be very difficult. In this paper, a high-rate real-time UA transceiver prototype we recently developed is presented. Together with carefully designed video processing algorithms, this prototype system is capable of real-time through-water video streaming. We implement our prototype system using the software-defined radio (SDR) technique, which enables a rapid system prototyping and testing without the cost of specialized hardware. This system consists of a universal software radio peripheral (USRP) interfaced with a pair of high-frequency transducers through a broadband impedance matching network designed in house. The transmitter and receiver signal processing algorithms are implemented using Python and run on external host computers. The cyclic prefix (CP)-based orthogonal frequency-division multiplexing (OFDM) technique [2]-[4] is used as the physical layer modulation scheme, together with a convolutional error-correction coding scheme.

There are several software-defined UA modems with very good performance for their applications [5], for example, the

GNU radio framework based systems [6], [7] and LabVIEW-based prototypes [8]-[10]. The system in [7] achieved a data rate of 260 kbps in real-time over a 200 m horizontal link. A highly directional high-rate reconfigurable UA modem has been demonstrated in [8]. A real-time UA OFDM system with adaptive modulation function has been developed in [10]. According to our best knowledge, our system is the first reported high-rate UA communication prototype design based on Python and USRP device, with real-time video streaming capability.

Both the transceiver and video coding designs are discussed in this paper. This system is tested in a UA communication experiment conducted in a hydroacoustic tank. We tested various combinations of modulation type and coding rate, leading to system data rates from 111 kbps to 445 kbps, which enables us to transmit real-time underwater video. Experimental results show that the system achieves a reliable bit-error-rate (BER) performance and high video transmission quality. Our system achieves a higher rate than the GNU radio framework based system [7] and the LabVIEW-based prototype [8].

The remainder of this paper is organized as follows. The model of a UA OFDM system is presented in Section II. The system link budget as well as the transmitter and receiver software implementation are shown in Section III. Video processing algorithms are presented in Section IV. The results of the tank experiment are shown in Section V. In Section VI, conclusions are drawn.

II. System Model

A frame-based coded UA OFDM system is proposed. In each OFDM block, a binary source bit stream is encoded and mapped into N_s data symbols drawn from either the phase-shift keying (PSK) or quadrature amplitude modulation (QAM) constellations. The N_s data symbols and N_p pilot symbols form an $N_c \times 1$ OFDM symbol vector $\boldsymbol{d}$, where N_c is the total number of subcarriers. The N_p pilot symbols are uniformly inserted into the OFDM symbol vector. The inverse discrete Fourier transform (IDFT) is performed on each OFDM symbol to convert the symbol to the time domain. Finally, a CP with a length of L_{cp} longer than the channel delay spread is prepended to the time domain OFDM symbol.

Carrier frequency	f_c	225 kHz
Sampling rate	R_s	500 kHz
Bandwidth	B	250 kHz
Number of subcarriers	N_c	16384
Subcarrier spacing	f_{sc}	15.2 Hz
Length of OFDM symbol	T	65.54 ms
Length of CP	T_{cp}	16.384 ms

At the receiver end, the USRP device downshifts and samples the received signal and passes the samples to the host computer. The receiver algorithm in the host computer down-samples the received samples and performs frame detection using the preamble sequence. When the frame is detected, the receiver starts processing the subsequent OFDM data blocks. After removing the CP, we obtain the baseband discrete time samples of one OFDM symbol as

$$\boldsymbol{r} = \boldsymbol{PF}^H \boldsymbol{Dh}_f + \boldsymbol{w} = \boldsymbol{PF}^H \boldsymbol{DFh}_t + \boldsymbol{w} \qquad (1)$$

where $\boldsymbol{F}$ is the $N_c \times N_c$ discrete Fourier transform (DFT) matrix with the (i,k)-th entry of $1/\sqrt{N_c}e^{-j2\pi(i-1)(k-1)/N_c}$, $i,k = 1,\ldots,N_c$, $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{d})$ is a diagonal matrix taking $\boldsymbol{d}$ as the main diagonal elements, $\boldsymbol{r}$ is the received signal vector, $\boldsymbol{w}$ is the noise vector, $\boldsymbol{h}_f$ is a vector containing the channel frequency response at all N_c subcarriers, $\boldsymbol{h}_t = \boldsymbol{F}^H\boldsymbol{h}_f$ is the discrete time domain representation of the channel impulse response, $\boldsymbol{P} = \mathrm{diag}(\boldsymbol{p})$, $\boldsymbol{p} = (1, e^{j2\pi f_o/B}, \ldots, e^{j2\pi(N_c-1)f_o/B})^H$ is the phase distortion caused by the frequency offset f_o, B is the bandwidth of the transmitted signal, and $(\cdot)^H$ denotes the conjugate transpose. Here we assume that the Doppler spread-induced signal waveform compression/expansion is resolved through resampling the received signal, and the remaining Doppler shift mainly causes signal phase distortion.

After estimating and removing the frequency offset, the frequency domain representation of the received signal can be written as

$$\boldsymbol{r}_f = \boldsymbol{Fr} = \boldsymbol{Dh}_f + \boldsymbol{w}_f \qquad (2)$$

where $\boldsymbol{w}_f = \boldsymbol{Fw}$ is the noise vector in the frequency domain.

III. TRANSCEIVER DESIGN

The USRP device from National Instruments (NI) is chosen as the SDR platform at the transmitter and receiver. Key parameters of this UA OFDM system are summarized in Table I, where we can observe two distinct features compared with most existing UA OFDM systems. Firstly, in order to increase the data rate, the system works in the ultrasonic band of 100 kHz to 350 kHz. Secondly, to improve the system spectral efficiency, a large number of subcarriers are used.

A. Link Budget

The carrier frequency of most existing UA communication systems is under 50 kHz. At the carrier frequency of $f_c = 225$ kHz, the absorption loss of signals in our system is higher than many existing systems. The transmission power required
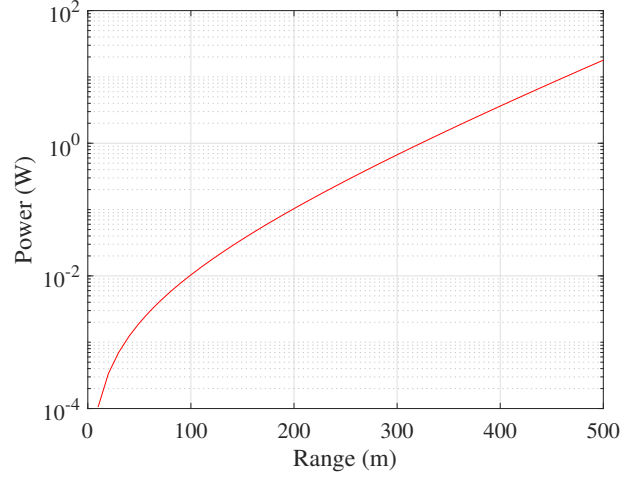

Fig. 1. Transmission power required versus communication range.

to achieve a 20 dB signal-to-noise ratio (SNR) at the receiver versus the communication range is shown in Fig. 1, where we apply the propagation loss model of UA communication channel in [11] and the noise model of UA systems in [12] to calculate the required transmission power. Parameters used in the calculation are listed in Table II, where efficiency refers to the ratio of the acoustic power emitted by the transducer to the electrical power drawn by the power amplifier.

Efficiency	0.16
Spreading factor	1.5
Shipping activity factor	1
Wind speed	11 m/s (40 km/h)

It can be seen that with 4 Watt transmission power, a communication range of 400 m can be reached. Note that in contract to convention UA communication systems which target long-range but low-rate applications, for many recent internet of underwater things (IoUT) applications, high-rate but short-to-medium range communications are of particular interest [13], [14].
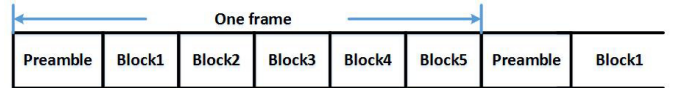

Fig. 2. Frame structure of the transmitted signals.

B. Transmitter Design

Fig. 2 shows the frame structure of the transmitted signals. Each frame contains $N_b = 5$ OFDM data blocks and one preamble block. The preamble block is an $N_c/2 = 8192$ long pseudo noise (PN) sequence followed by $N_c/2$ zeros and is used for synchronization and frame detection. Among

TABLE III
UA OFDM SYSTEM DATA RATES AND BER PERFORMANCE

Modulation	Coding rate	Frame length	Data rate	BER
QPSK	1/2	10574 bits	111.27 kbps	0
QPSK	3/4	15864 bits	166.94 kbps	0
QPSK	1	21160 bits	222.67 kbps	0.16%
16-QAM	1/2	21154 bits	222.61 kbps	0.02%
16-QAM	3/4	31734 bits	333.95 kbps	0.93%
16-QAM	1	42320 bits	445.35 kbps	0.76%

Fig. 3. Receiver flowchart.

Fig. 4. Receiver baseband processing block diagram.

the $N_c = 16384$ subcarriers, there are $N_s = 10580$ data subcarriers and $N_p = N_c/4 = 4096$ uniformly spaced pilot subcarriers. The data symbols are modulated by quadrature PSK (QPSK) or 16-QAM constellations. The source bits are encoded by the convolutional codes with coding rate of $1/2$, $3/4$, or 1 (coding rate 1 means the encode/decode process is bypassed). The number of source bits in one OFDM frame and the corresponding data rates are shown in Table III.

C. Receiver Design

The flowchart of the receiver signal processing is shown in Fig. 3. In the detection mode, the receiver searches the frame head and remains in this mode if it fails to detect the frame head. Otherwise, the system enters the decoding mode where the frame payload is received and processed. At the end of one detected frame the receiver returns to the detection mode again and starts searching the next frame.

Cross-correlation of the baseband signal and the local synchronization sequence is performed in the detection mode. By checking the cross-correlation results, the receiver detects the received synchronization sequence which is the frame head. After detecting the frame head, the receiver starts processing the subsequent data payload. Each OFDM block is received through the NI USRP 2920 device and then downsampled after passing through a low-pass filter followed by the CP removal. The receiver then processes the received baseband signal to obtain the decoded bits. After processing one OFDM block, the receiver checks if all data in one frame has been received. It enters the detection mode again if all OFDM blocks in one frame have been processed. Otherwise, it starts processing the next OFDM block.

1) OFDM Baseband Processing: The baseband processing is shown in Fig. 4. The receiver first performs frequency offset estimation and compensation for each OFDM symbol using the null subcarriers. Then the baseband signal is passed through channel estimation, where the least-squares method is used to estimate the UA channel frequency response on the pilot subcarriers. Linear interpolation algorithm is then adopted to estimate the data subcarrier channel response via the pilot subcarrier channel response. The estimated channel response is used to equalize the received data subcarriers. Soft demodulation operation is performed to the equalized data subcarriers leading to a soft bit sequence. This soft bit sequence is finally passed into the soft-input Viterbi decoder.

2) Frame Searching: In each attempt of the frame head searching, the system receives $N_c/2$ samples and attaches
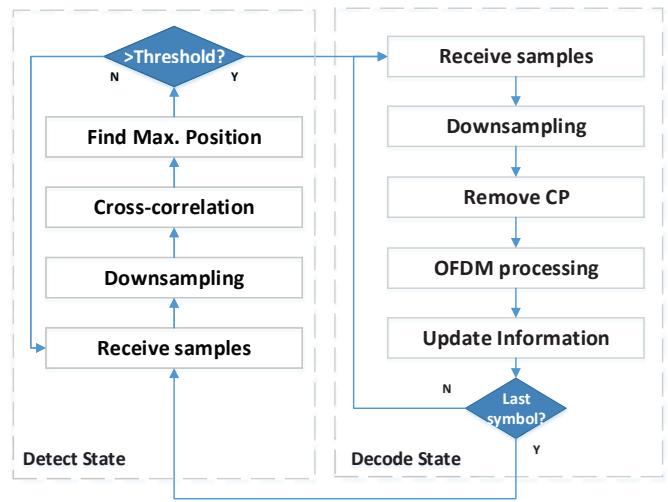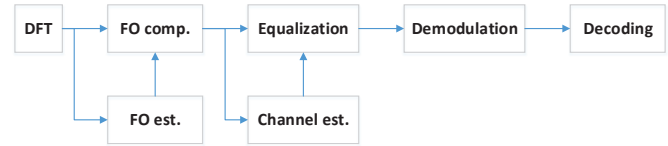
them to the $N_c/2$ samples received in the previous attempt, leading to an N_c sequence. This sequence and the local synchronization sequence are passed through a DFT-based cross-correlator. If the quotient of the maximum correlation over the average correlation is larger than the preset threshold and at the same time the position of the maximum point is within the first $N_c/2$ samples, the system takes this position as the frame head. Otherwise, the system starts the next attempt.

3) Frequency Offset Removing: The Doppler shift compensation algorithm used in [15] is adopted here to estimate and remove the carrier frequency offset. This module calculates the tentative frequency points according to a predetermine setting and generates an N_c-long phase rotation sequence for each frequency point. After receiving a new OFDM symbol, for each tentative frequency point, this module compensates the frequency offset on the received OFDM symbol using the corresponding phase rotation sequence and then calculates the average power of the null subcarriers. The system then estimates the frequency offset by comparing the average power results of all the tentative frequency points and uses the phase rotation sequence associated with the minimal power on null subcarriers to compensate the frequency offset.

IV. VIDEO CODING DESIGN

In this section, we present the video coding and decoding algorithms for through-water video streaming. A conventional approach for real-time image and video communication is to first compress the images captured by the video camera and then transmit the compressed images independently. This
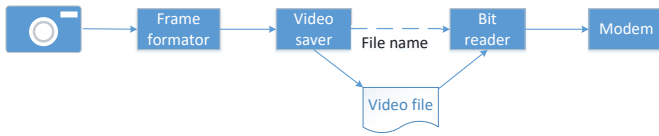
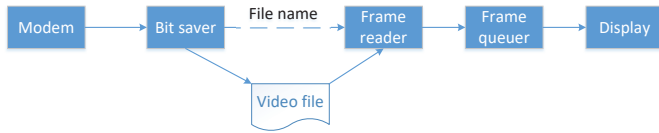Fig. 5. Transmitter video processing block diagram.

Fig. 6. Receiver video processing block diagram.

Fig. 7. Experimental system setup.

approach has a short image processing delay but requires the modem to support a high data rate, which is not suitable for the bandwidth limited UA communication.

To solve this problem, we propose to encode the captured images into video files by removing the time redundancy among images. Thus, transmitting these video files requires a lower data rate than transmitting the images. By increasing the number of images in one video file, we can increase the video compression ratio, which reduces the data rate required to transmit the images. However, this increases the processing delay of real-time video communication. Therefore, it is important to carefully design the duration parameter to achieve a balance between the data rate and processing delay.

The block diagrams of video processing at the transmitter and receiver are shown in Fig. 5 and Fig. 6, respectively. At the transmitter side, images generated by the video camera are captured with a predefined frame rate and formatted into a designed resolution. These frames are saved into a video file with the H.265 coding. After a predetermined number of frames are saved, the video saver passes the file name to the bit reader, which reads bytes from the video file and attaches the starter and termination flags to the byte sequence. The byte sequence is converted into a bit sequence which can be processed and transmitted by the OFDM-based high-rate UA modem we develop.

At the receiver end, the UA modem processes the received signal and passes the decoded bit sequence to the bit saver module. The bit saver firstly converts the bit sequence into a byte sequence and searches the starter flag. Once the starter flag is found, the bit saver starts saving the consequent bytes into a video file. Meanwhile, this module searches the termination flag. Once the termination flag is found, the bit saver stops saving data and passes the file name to the frame reader, which reads image frames from the video file and passes all the frames into the frame queue. Finally, the frame queuer reads frame data from the queue with a predefined frame rate and displays the video information.

V. EXPERIMENT RESULTS

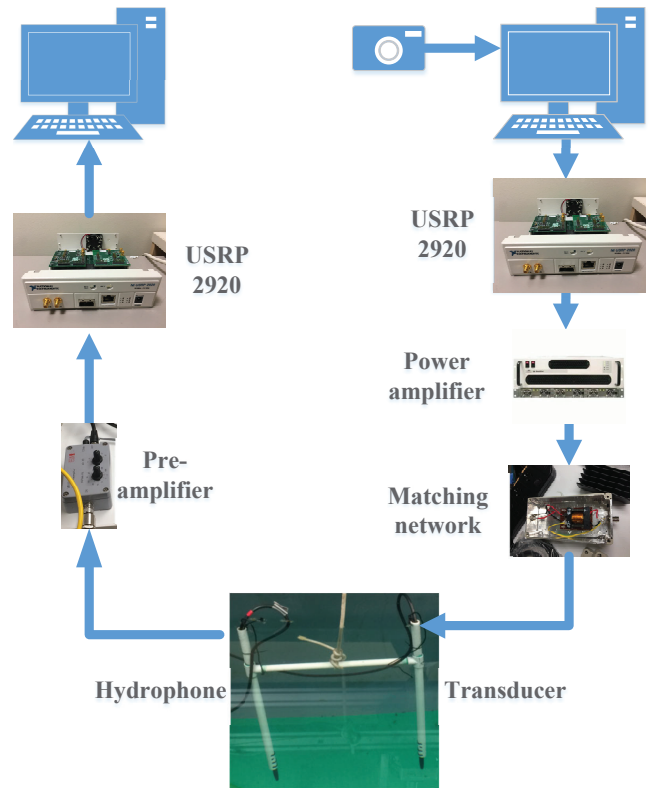In this section, we study the performance of the proposed Python and USRP-based high-rate UA OFDM system in real-time video streaming. The experimental system setup is shown in Fig. 7. Each NI USRP 2920 device is connected to a Linux desktop computer where the USRP driver and Python 3.8 are installed for signal processing. A video camera is connected to the transmitter computer. A Reson 4034 transducer is connected through a power amplifier (Tomco BT-AlphaA) and a broadband impedance matching network (designed in house) to the LFTX daughter board mounted on one USRP 2920 device for transmitting acoustic signals. At the receiver end, the LFRX daughter board mounted on the other USRP 2920 device is connected to a Reson 4034 hydrophone through a pre-amplifier (Reson VP1000) for the acquisition of real-time acoustic signals.

The system we developed was tested in a hydroacoustic tank at Curtin University. The transducer and the hydrophone were attached to an H-shaped plastic jig, and the distance between them is around 0.5 m. The amplitude of a typical normalized channel impulse response between the transducer and the hydrophone during the experiment estimated by the pilot subcarriers is shown in Fig. 8. It can be seen that the maximal channel delay spread is around 6 ms which is shorter than the length of the CP. This verifies that the system parameters are correctly chosen. Fig. 9 illustrates the scatter plot of the received symbols in one OFDM block after channel equalization. It can be seen that after the channel equalization, most of the 16-QAM symbols are properly aggregated into the normalized modulation constellations.
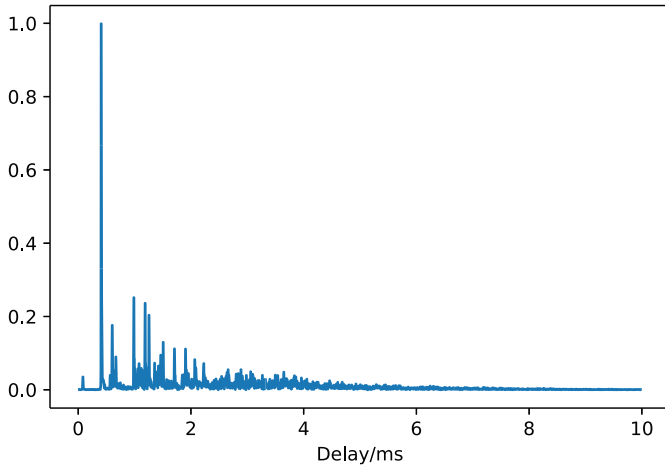
Fig. 8. Amplitude of normalized channel impulse response estimated by the pilot subcarriers.
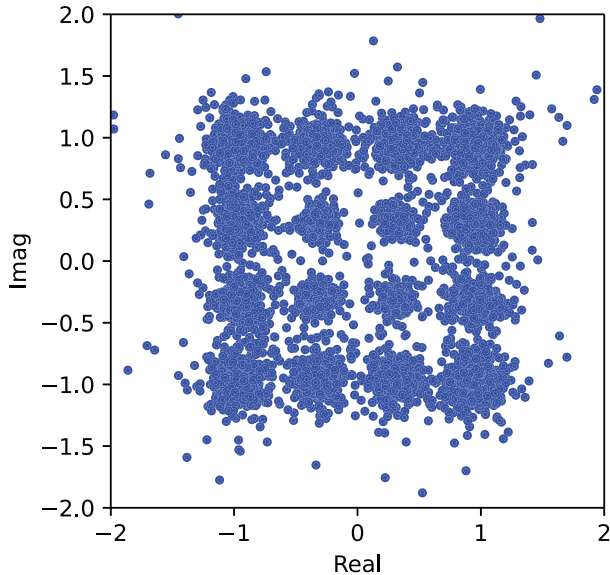

Fig. 9. Scatter plot of the equalized symbols of one OFDM block.

The system BER performance is shown in Table III for various modulation and coding rate combinations. The results show that the proposed system can achieve very low BER. This system is used to transmit real-time video with a resolution of 640×480 and refresh rate of 10 frames/second. Each video file recorded 50 frames of images, which corresponds to 5 seconds of video data. From the link in [16], we can see that the real-time video transmission has a high quality apart from a short delay introduced by video compression as explained in Section IV. The system can be configured to support a higher resolution by reducing the refresh rate.

VI. CONCLUSIONS

In this paper, an NI USRP and Python-based implementation of high-rate real-time UA OFDM system has been presented. Tank tests showed that the system achieves a data rate of 445 kbps with a single transducer and is capable of real-time video streaming. We are working on improving the Doppler shift and Doppler spread estimation and compensation algorithms such that this high-rate system has a robust performance in practical environment with moving communication platforms.

REFERENCES

[1] D. Kilfoyle and A. Baggeroer, "The state of the art in underwater acoustic telemetry," *IEEE J. Ocean. Eng.*, vol. 25, no. 1, pp. 4-27, Jan. 2000.
[2] P. Chen, Y. Rong, S. Nordholm, Z. He, and A. Duncan, "Joint channel estimation and impulsive noise mitigation in underwater acoustic OFDM communication systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 6165-6178, Sep. 2017.
[3] P. Chen, Y. Rong, S. Nordholm, and Z. He, "Joint channel and impulsive noise estimation in underwater acoustic OFDM systems," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10567-10571, Nov. 2017.
[4] S. Wang, Z. He, K. Niu, P. Chen, and Y. Rong, "New results on joint channel and impulsive noise estimation and tracking in underwater acoustic OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2601-2612, Apr. 2020.
[5] H. S. Dol, P. Casari, T. van der Zwan, and R. Otnes, "Software-defined underwater acoustic modems: Historical review and the NILUS approach," *IEEE J. Ocean. Eng.*, vol. 42, no. 3, pp. 722-737, Jul. 2017.
[6] E. Demirors, G. Sklivanitis, T. Melodia, S. N. Batalama, and D. A. Pados, "Software-defined underwater acoustic networks: Toward a high-rate real-time reconfigurable modem," *IEEE Commun. Magazine*, vol. 53, no. 11, pp. 64-71, Nov. 2015.
[7] E. Demirors, G. Sklivanitis, G. E. Santagati, T. Melodia, and S. N. Batalama, "A high-rate software-defined underwater acoustic modem with real-time adaptation capabilities," *IEEE Access*, vol. 6, pp. 18602-18615, 2018.
[8] L. E. Emokpae, S. E. Freeman, G. F. Edelmann, and D. M. Fromm, "Highly directional multipath free high data-rate communications with a reconfigurable modem," *IEEE J. Ocean. Eng.*, vol. 44, no. 1, pp. 229-239, Jan. 2019.
[9] P. Chen, Y. Rong, S. Nordholm, and Z. He, "An underwater acoustic OFDM system based on NI CompactDAQ and LabVIEW," *IEEE Systems Journal*, vol. 13, no. 4, pp. 3858-3868, Dec. 2019.
[10] S. Barua, Y. Rong, S. Nordholm, and P. Chen, "Real-time adaptive modulation schemes for underwater acoustic OFDM communication," *MDPI Sensors*, vol. 22, no. 3436, Apr. 2022.
[11] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic communication channel," in *Proc. WUWNet*, Los Angeles, CA, USA, Sep. 25, 2006.
[12] R. Coates, *Underwater Acoustic Systems*. New York: Wiley, 1989.
[13] F. Sun, X. Zhu, Y. Xue, J. Li, and Y. R. Zheng, "Front-end circuits for ultra-high-frequency underwater acoustic communication systems," in *Proc. MTS/IEEE Oceans*, Hampton Roads, Virginia, USA, Oct. 17-20, 2022.
[14] A. Jarrot *et al.*, "High-speed underwater acoustic communication for multi-agent supervised autonomy", in *Proc. Underwater Commun. Networking*, Lerici, Italy, Sep. 2021.
[15] H. Yan, L. Wan, S. Zhou, Z. Shi, J. H. Cui, J. Huang, and H. Zhou, "DSP based receiver implementation for OFDM acoustic modems," *Phys. Commun.*, vol. 5, no. 1, pp. 22-32, 2012.
[16] High-rate underwater acoustic communication system capable of real-time video transmission, https://ddfe.curtin.edu.au/yurong/.