A Software-defined Optical Wireless OFDM System for Underwater Video Communication

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

OCEANS 2022, Hampton Roads | 978-1-6654-6809-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/OCEANS47191.2022.9977077

***Abstract*—In this paper, we present a software-defined underwater optical wireless communication system that we recently developed, which is capable of real-time through-water video streaming. The orthogonal frequency-division multiplexing technique is used for signal modulation. Using the software-defined radio technique, the transceiver signal processing and video processing algorithms are implemented using Python and run on external host computers. Experimental results show that this system can achieve very low error rate and transmit real-time video with a high quality.**

***Index Terms*—Orthogonal frequency-division multiplexing, software-defined radio, underwater optical wireless communication.**

I. Introduction

Recently, there is an increasing interest in reliable and high-rate underwater wireless communication, due to a surge of underwater applications involving the interconnection of underwater vehicles and sensor networks. In general, underwater wireless communication can be implemented through three physical transmission approaches: radio-frequency (RF), acoustic, and optical. Considering the limited bandwidth of the acoustic channel and the large attenuation of radio wave in water, underwater optical wireless communication (UOWC) has become an attractive and viable approach for short and moderate distance underwater communication [1], [2].

Duntley found that light with wavelength from 450 nm to 550 nm experiences a low attenuation in seawater, which corresponds to the blue and green spectrum [3]. This property establishes the foundation for the development of UOWC. By exploiting the low absorption in the blue-green spectrum, UOWC systems have the potential to support high-rate links and transmit a large amount of data in real time required by many underwater applications. A recent numerical study shows that it is possible to achieve light emitting diode (LED)-based visible light communication over a distance of 500 m in pure seawater by employing a single photon avalanche diode [4]. The BlueComm UOWC system supports 20 Mbps underwater data transmission over a distance of 200 m [5]. The Ambalux UOWC system provides a 10 Mbps data transmission with a range of 40 m [6].

This research was supported by the Defence Science Centre, an initiative of the State Government of Western Australia.

Orthogonal frequency-division multiplexing (OFDM) is used extensively in broadband wired and wireless communication systems thanks to its high spectral efficiency and its capability in mitigating inter-symbol interference with a low computational complexity [7], [8]. This technology also attracts attention in UOWC systems in recent years. An intensity modulation/direct detection (IM/DD)-OFDM UOWC system using a 405 nm blue laser diode achieves a data rate of 1.45 Gbps over a distance of 4.8 m [9]. An OFDM UOWC system with 16-quadrature amplitude modulation (QAM) employing TO-9 packaged fiber-pigtailed laser diode demonstrates the capability of 4.8 Gbps transmission rate over a 5.4 m range in water tank [10].

The software-defined radio (SDR) technique is a powerful tool for designing physical layer reconfigurable and spectrum efficient communication systems while avoiding the cost of specialized hardware. It has received a lot of attention in RF communications as well as in the underwater communication research community [11]. The flexibility provided by SDR suits the rapidly time-varying underwater communication channels. In [12], a universal software radio peripheral (USRP) platform integrated with GNU based host computer is applied to evaluate the performance of a direct current offset (DCO)-based OFDM UOWC system.

In this paper, we present a UOWC system that we recently developed, which is capable of real-time through-water video streaming. The OFDM technique is used for signal modulation. Using the SDR technology, the transceiver signal processing and video processing algorithms are implemented using Python and run on external host computers. Experimental results show that this system can achieve a low error rate and transmit real-time video with a high quality.

The rest of this paper is organized as follows. The model of an OFDM-based UOWC system is presented in Section II. The transmitter and receiver software implementation is 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

In this paper, a frame-based coded DCO-OFDM communication system implemented by Python and the National Instruments (NI) USRP device is proposed. In each OFDM block, a binary source data stream of L_b bits is encoded into a sequence of L_c bits. The encoded bits are mapped into N_s data symbols drawn from either the phase-shift keying (PSK) or QAM constellations. Then the above N_s data symbols together with N_p pilot symbols are mapped into an $N_c \times 1$ OFDM symbol vector $\boldsymbol{d}$ at the transmitter, where N_c is the total number of subcarriers. These N_p quadrature PSK (QPSK) modulated 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 cyclic prefix (CP) with a length of L_{cp} longer than the channel delay spread is prepended to the time domain OFDM symbol.

After digital-to-analog conversion, the signal is sent to an optical transmitter front-end, which adds DC bias, converts the electrical signal to blue LED light, and transmits the resulting signal through the optical channel. At the receiver front-end, it is assumed that the DC offset is perfectly removed. The output of the optical receiver is passed to a USRP device, which performs analog-to-digital conversion and passes the signal samples to the host computer. The receiver algorithm in the host computer downsamples 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

$$\begin{aligned}\boldsymbol{r} &= \boldsymbol{F}^H \boldsymbol{D} \boldsymbol{h}_f + \boldsymbol{w} \\ &= \boldsymbol{F}^H \boldsymbol{D} \boldsymbol{F} \boldsymbol{h}_t + \boldsymbol{w} \end{aligned} \tag{1}$$

where $\boldsymbol{F}$ is an $N_c \times N_c$ 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, $\boldsymbol{r} = (r[1],\ldots,r[N_c])^T$ is the received signal vector, $\boldsymbol{w} = (w[1],\ldots,w[N_c])^T$ is the noise vector, $\boldsymbol{h}_f = (h_f[1],\ldots,h_f[N_c])^T$ 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 with a maximum delay of L_m, $(\cdot)^T$ and $(\cdot)^H$ denote the transpose and the conjugate transpose, respectively. The frequency domain representation of the received signal can be written as

$$\begin{aligned}\boldsymbol{r}_f &= \boldsymbol{F}\boldsymbol{r} \\ &= \boldsymbol{F}\boldsymbol{F}^H \boldsymbol{D} \boldsymbol{h}_f + \boldsymbol{F}\boldsymbol{w} \\ &= \boldsymbol{D}\boldsymbol{h}_f + \boldsymbol{w}_f \end{aligned} \tag{2}$$

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

III. System Design

There are a variety of commercially available SDR products that feature a mixture of software and hardware configurations at different cost, such as USRP, HackRF, BladeRF, and RTL-SDR. The USRP device utilized in this paper is a radio front-end launched by Ettus Research that provides a wider frequency range and dedicated hardware devices to support different applications. A well designed USRP driver is launched by Ettus Research to enable the communication between their products and the host processing units. This driver supports C++ interface as well as a wrapped Python version.

Python is one of the top coding languages due to its efficient nature of programming features. Since Python is free to the public, this leads to a rapidly growing user community. Python codes are easy to learn and to read, thanks to the built-in libraries and debugging feature. The comprehensive and powerful libraries are structured to focus on general programming and contains OS specific, threading, networking, and I/O modules, which enable developers to achieve their ultimate goal in an elegant and well-organized way. Python has also simplified debugging for programmers due to its built-in debugging feature.

A large number of popular third-party libraries are available in the Python community for vectorized computation on the CPU, which makes Python suitable for computation-intensive applications. These libraries include Numpy, Numba, and SciPy. Numpy contains a set of commonly used functions for performing many vectorized and linear algebra operations in a similar way to MATLAB. These libraries are implemented by well-optimized C code, which enables developers to utilize the flexibility of Python as well as the speed of compiled code. Numba extends the capability of Numpy by allowing users to define vector functions and precompiling part of codes to the assembly language and thus further improves the execution speed. Python code can also be easily extended to GPUs using the CuPy library.

Based on the advantages above, we choose to implement the UOWC system using USRP and Python. Key parameters of this UOWC OFDM system are summarized in Table I, where we can observe that compared with most existing OFDM systems, our UOWC system has a large number of subcarriers. In the following, we present the system software design at the transmitter and the receiver.

TABLE I
UOWC OFDM System Parameters

Carrier frequency	f_c	3.5 MHz
Sampling rate	R_s	2.5 MHz
Bandwidth	B	1.25 MHz
Number of subcarriers	N_c	16384
Subcarrier spacing	f_{sc}	76.3 Hz
Length of OFDM symbol	T	13.1 ms
Length of CP	T_{cp}	3.3 ms

A. Transmitter Design

Fig. 1 shows the frame structure of the transmitted signals. It can be seen that each data frame contains five OFDM data blocks and one preamble block. The preamble block consists of an 8192 long pseudo noise (PN) sequence followed by 8192 zeros and is used for synchronization and frame detection. Among the $N_c = 16384$ subcarriers, there are 10580 data subcarriers and $N_c/4 = 4096$ uniformly spaced pilot

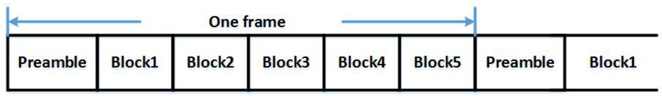

Fig. 1. Frame structure of the transmitted signals.

subcarriers. The data symbols are modulated by QPSK or 16-QAM constellations. The source bits are encoded by the convolutional codes with coding rate of $1/2$, $3/4$, or 1 (no coding). The number of source bits in each OFDM block and the corresponding data rates are shown in Table II.

TABLE II
UOWC OFDM SYSTEM DATA RATES

Modulation	Coding rate	Bit length	Data rate
QPSK	$1/2$	10574	556.4 kbps
QPSK	$3/4$	15864	834.7 kbps
QPSK	1	21160	1.1 Mbps
16-QAM	$1/2$	21154	1.1 Mbps
16-QAM	$3/4$	31734	1.67 Mbps
16-QAM	1	42320	2.23 Mbps

B. Receiver Design

The flowchart of the receiver signal processing is shown in Fig. 2. The receiver acquires samples from the NI USRP 2920 device. The samples are passed through a low-pass filter and downsampled. 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.

The search of the frame head is performed by cross-correlating the baseband signal with the local synchronization sequence. By checking the cross-correlation results, the receiver detects the received synchronization sequence which is the frame head. When the frame is detected, the receiver starts processing the subsequent OFDM data blocks, which will be discussed later to obtain the decoded bits. The decoded bits are compared with the transmitted bit sequence to calculate the bit-error-rate (BER). System counters including the detected frame counter, the OFDM symbol counter, and the successfully decoded OFDM symbol counter, are updated. After processing one OFDM block, the receiver checks if all the data is one frame has been received. It enters the detection mode again if all the OFDM blocks have been processed. Otherwise, it starts processing the next OFDM block.

1) OFDM Baseband Processing: The receiver baseband processing is shown in Fig. 3. It can be seen that the baseband signal is passed through channel estimation, which uses the least-squares method to estimate the UOWC channel frequency response on the pilot subcarriers. Linear interpolation is adopted to estimate the data subcarrier channel response via the pilot subcarrier channel response. Then the estimated channel is used to equalize the received data subcarriers. Soft demodulation is performed to the equalized data subcarriers leading to a soft bit sequence, which is finally passed to the soft-input Viterbi decoder.

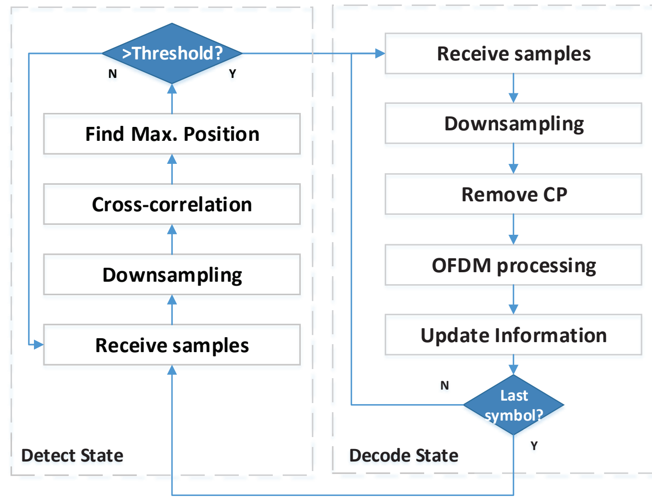

Fig. 2. Receiver flowchart.

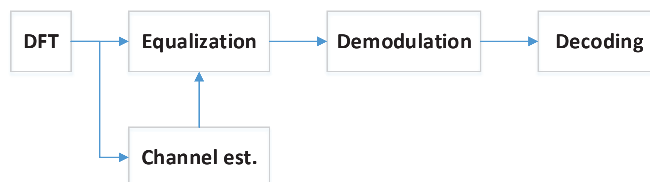

Fig. 3. Receiver baseband processing block diagram.

2) Frame Searching: DFT-based cross-correlation between the received samples and the local synchronization sequence is performed to detect the frame head. In each attempt, the system receives N_{PN} samples and attaches them to the N_{PN} samples received in the previous attempt, leading to a $2N_{PN}$ sequence. This sequence and the local synchronization sequence are passed through a $(2N_{PN}+2)$-point fast Fourier transform (FFT). The corresponding points of the two resulting sequences are multiplied together followed by the inverse fast Fourier transform (IFFT). The system then calculates the average power of the IFFT output and finds out the point with the maximum power. If the quotient of the maximum power over the average power is larger than the preset threshold and at the same time the position of the maximum power point is within the first N_{PN} samples, the system takes this position as the frame head. Otherwise, the system starts the next attempt.

IV. VIDEO CODING DESIGN

The block diagrams of video processing at the transmitter and receiver are shown in Fig. 4 and Fig. 5, respectively. At the transmitter, images captured by the video camera 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 bits from the video file and attaches the starter and termination flags to the bit sequence.

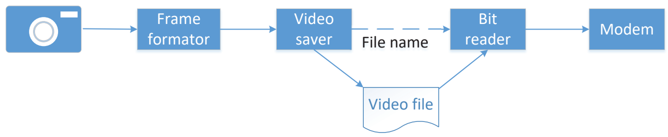

Fig. 4. Transmitter video processing block diagram.

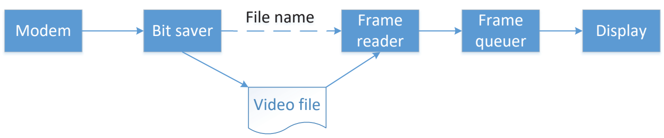

Fig. 5. Receiver video processing block diagram.

At the receiver, the decoded bit sequence is passed to the bit saver, which searches the starter flag. Once it is found, the bit saver starts saving the consequent bytes into a video file. Meanwhile, this module searches the termination flag. Once it 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 UOWC OFDM system. The experimental system setup is shown in Fig. 6. The prototype system consists of two NI USRP 2920 devices with LFTX/LFRX daughterboards, two host Linux desktop computers, and a pair of Hyperion LiFi transmitter and receiver front-ends. Each USRP device is connected to a Linux desktop computer where the USRP driver and Python 3.8 are installed for signal processing. The LiFi transmitter is connected to the LFTX daughter board mounted on one USRP device for transmitting real-time signals carried by blue LED light. The light was transmitted to the receiver through a glass water tank. At the receiver end, the LFRX daughter board mounted on the other USRP device is connected to the LiFi receiver front-end for the acquisition of real-time signals.

Fig. 7 shows the scatter plot of received symbols in one OFDM block after channel equalization. We can see that most of the QPSK symbols are properly aggregated into the normalized modulation constellations. This system is used to transmit real-time video with a resolution of 640×480 and refresh rate of 10 frames/s. From the link in [13], we can see that the video transmission has a high quality apart from a 0.5 s delay introduced by video compression (5 frames in one video file).

VI. Conclusions

In this paper, a software-defined UOWC system based on USRP and Python is presented. Experimental results show that this system can achieve a low error rate and transmit real-time video with a high quality.

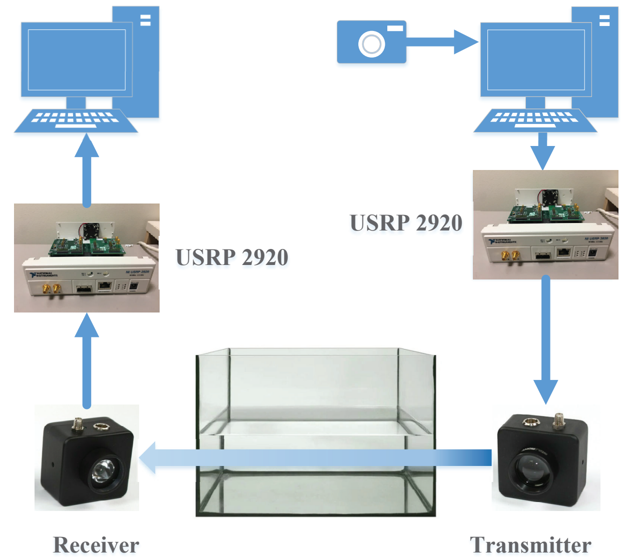

Fig. 6. Experimental system setup.

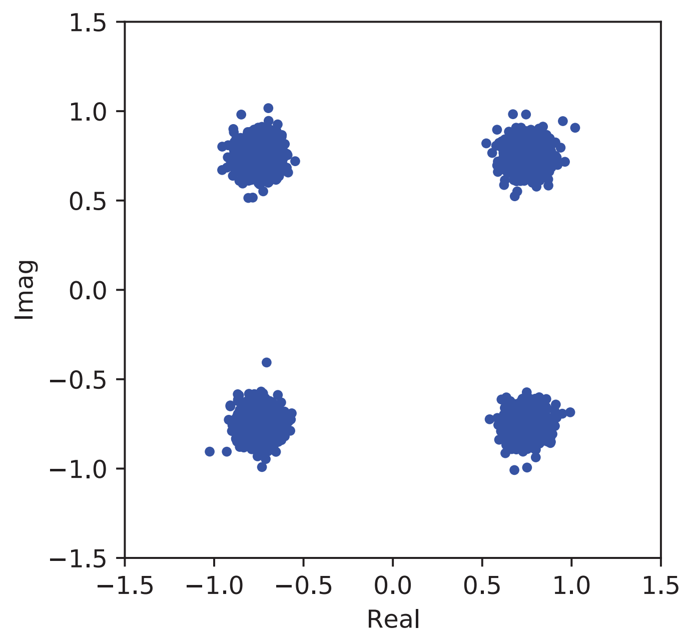

Fig. 7. Scatter plot of the equalized symbols of one OFDM block.

References

[1] G. Baiden, Y. Bissiri, and A. Masoti, "Paving the way for a future underwater omni-directional wireless optical communication systems," *Ocean Eng.* vol. 36, pp. 633-640, July 2009.

[2] Y. Rong, S. Nordholm, and A. Duncan, "On the capacity of underwater optical wireless communication systems," in *Proc. Underwater Commun. Networking*, Lerici, Italy, Sep. 2021.

[3] S. Q. Duntley, "Light in the sea," *J. Opt. Soc. Amer.*, vol. 53, pp. 214-233, Feb. 1963.

[4] C. Wang, H.-Y. Yu, and Y.-J. Zhu, "A long distance underwater visible light communication system with single photon avalanche diode," *IEEE Photon. J.*, vol. 8, no. 5, Oct. 2016.

[5] S. Fasham and S. Dunn, "Developments in subsea wireless communications," in *Proc. IEEE Underwater Technol.*, Chennai, India, 2015.

[6] M.-A. Khalighi *et al.*, "Underwater wireless optical communication: Recent advances and remaining challenges," in *Proc. Int. Conf. Transparent Opt. Netw.*, Graz, Austria, 2014.

[7] M. Chitre, S. H. Ong, and J. Potter, "Performance of coded OFDM in very shallow water channels and snapping shrimp noise," in *Proc. MTS/IEEE OCEANS*, Washington, DC, Sep. 2005.

[8] 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, pp. 2601-2612, Apr. 2020.

[9] K. Nakamura, I. Mizukoshi, and M. Hanawa, "Optical wireless transmission of 405 nm, 1.45 Gbit/s optical IM/DD-OFDM signals through a 4.8 m underwater channel," *Opt. Exp.*, vol. 23, no. 2, pp. 1558-1566, 2015.

[10] H. M. Oubei *et al.*, "4.8 Gbit/s 16-QAM-OFDM transmission based on compact 450-nm laser for underwater wireless optical communication," *Opt. Exp.*, vol. 23, no. 18, pp. 23302-23309, 2015.

[11] H. Luo, K. Wu, R. Ruby, Y. Liang, Z. Guo, and L. M. Ni, "Software-defined architectures and technologies for underwater wireless sensor networks: A survey," *Communications Surveys & Tutorials IEEE*, vol. 20, no. 4, pp. 2855-2888, 2018.

[12] W. C. Cox, J. A. Simpson, and J. F. Muth, "Underwater optical communication using software defined radio over LED and laser based links," in *Proc. IEEE MILCOM*, Nov. 2011, pp. 2057-2062.

[13] Y. Rong, Demo of real-time video transmission through an underwater optical wireless communication system, https://ddfe.curtin.edu.au/yurong/.